

Concurrent Single-Label Image Classification and Annotation via Efficient Multi-Layer Group Sparse Coding

Gao, Shenghua; Chia, Liang-Tien; Tsang, Ivor Wai-Hung; Ren, Zhixiang

2014

Gao, S., Chia, L.-T., Tsang, I. W.-H., & Ren, Z. (2014). Concurrent Single-Label Image Classification and Annotation via Efficient Multi-Layer Group Sparse Coding. *IEEE Transactions on Multimedia*, 16(3), 762-771.

<https://hdl.handle.net/10356/81696>

<https://doi.org/10.1109/TMM.2014.2299516>

© 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: [<http://dx.doi.org/10.1109/TMM.2014.2299516>].

Downloaded on 25 Aug 2022 18:00:50 SGT

Concurrent Single-Label Image Classification and Annotation via Efficient Multi-Layer Group Sparse Coding

Shenghua Gao, Liang-Tien Chia, Ivor Wai-Hung Tsang, and Zhixiang Ren

Abstract—We present a multi-layer group sparse coding framework for concurrent single-label image classification and annotation. By leveraging the dependency between image class label and tags, we introduce a multi-layer group sparse structure of the reconstruction coefficients. Such structure fully encodes the mutual dependency between the class label, which describes image content as a whole, and tags, which describe the components of the image content. Therefore we propose a multi-layer group based tag propagation method, which combines the class label and subgroups of instances with similar tag distribution to annotate test images. To make our model more suitable for nonlinear separable features, we also extend our multi-layer group sparse coding in the Reproducing Kernel Hilbert Space (RKHS), which further improves performances of image classification and annotation. Moreover, we also integrate our multi-layer group sparse coding with k NN strategy, which greatly improves the computational efficiency. Experimental results on the LabelMe, UIUC-Sports and NUS-WIDE-Object databases show that our method outperforms the baseline methods, and achieves excellent performances in both image classification and annotation tasks.

Index Terms—sparse coding, image classification, image annotation, kernel trick

I. INTRODUCTION

Image classification and image annotation are two classical problems in computer vision. Given an image, image classification tells people what is the theme of the image (high-level semantic meaning), and image annotation tells people what objects are inside the image and their properties (tags for image component descriptions). This paper targets single-label image classification where each image only belongs to one class.

Lots of image classification [23][40] and image annotation [43][55] frameworks have been developed in recent years. However, most of these frameworks solve the image classification or image annotation independently. As we know, the high-level semantic meaning of an image can help the prediction of objects in this image, and image components can jointly help predict the semantic class label of this image. For example,

Shenghua Gao is with Advanced Digital Sciences Center, Singapore. Liang-Tien Chia, Ivor Wai-Hung Tsang and Zhixiang Ren are with School of Computer Engineering, Nanyang Technological University, Singapore.

Email: {shenghua.gao}@adsc.com.sg

This study is partially supported by the research grant for the Human Sixth Sense Programme at the Advanced Digital Sciences Center from Singapore’s Agency for Science, Technology and Research (A*STAR).

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

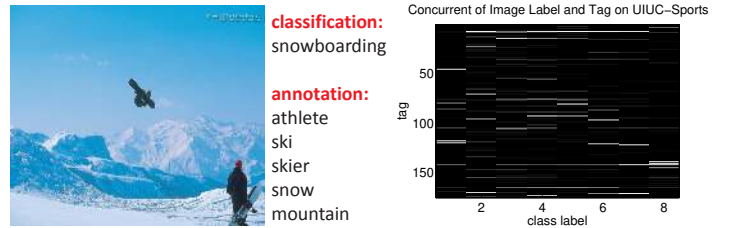


Fig. 1. An illustration of image tags, class label, and their concurrence map on UIUC-Sports dataset. For UIUC-Sports, there are 8 classes, and 175 tags. The concurrence map shows that usually a tag only appears in one or very few classes, so we can predict the tag based class label, and vice versus.

in Fig. 1, the class label “snowboarding” and tags “snow”, “ski”, “athlete”, etc. can reciprocally boost the prediction of each other. To further illustrate the relationship between image class label and tags, we also show the concurrence of the class label and tags on UIUC-Sports dataset in Fig. 1. The concurrence map shows that usually a tag only appears in one or very few classes, so we can predict the tag based class label, and vice versus. Such dependency between class label and tags shows that the image classification and image annotation are closely related, which motivates us to solve image classification and image annotation problems concurrently. Meanwhile, from the image users’ perspective, concurrent image classification and annotation enhance the users’ understanding of the image content. Fig. 1 also illustrates a possible application of such concurrent image classification and image annotation: someone takes lots of images in different activities, like snowboarding, skiing, swimming, etc. By using image classification, images belonging to different activities can be automatically sorted out. Then by using image annotation technique, each image can be annotated with different tags which further describe the image contents, we can group images based on a high-level semantic concept, and each tag which may be a name of object in the image, like “snow”, “athlete”, is used to describe the image contents.¹

Recently sparse coding has demonstrated good performance for single-label image classification [47][53][50] under the assumption that “If sufficient training samples are available from each class, it would be possible to represent the test samples

¹As for the multi-label image classification task on, for example, PASCAL VOC datasets, it is a different application scenario. On PASCAL VOC datasets, multiple objects appear in the same image, like car, person, bike. No higher-level concepts (like scene or activities concepts) are given to group these images. Therefore such task is different from the task we are dealing with.

as a linear combination of those training samples from the same class [47]". Moreover, sparse coding also demonstrates good performance for image annotation task [45][26] with a label transfer strategy.

Motivated by the success of sparse coding for single-label image classification and image annotation, as well as the close relationship between image classification and image annotation, in this paper, we present a multi-layer group sparse coding framework. Such a framework not only inherits the ability of sparse coding in single-label image classification, but also encodes additional prior information, including the dependency between the class label and tags, and group sparsity of the reconstruction coefficients (sparse codes) corresponding to the instances with the same class label, *etc.* The architecture of our multi-layer group structure is depicted in Fig. 2. Specifically, our multi-layer group sparse architecture contains three layers: the instance layer, the class-based group layer and the tag-based subgroup layer. On the class-based group layer, the sparse codes corresponding to the instances with the same class label form a class-based group. On the tag-based subgroup layer, the sparse codes corresponding to the instances with both the same class label and similar tag distribution form a tag-based subgroup. The sparsity on these three layers implies the minimal number of the instances, class-based groups and tag-based subgroups are used for reconstructing a test image. Based on the reconstruction error for each class-based group and tag-based subgroup, we can classify and annotate the test image concurrently.

The contributions of this paper can be summarized as follows. Firstly, we present a multi-layer group sparse coding framework to solve the image classification and image annotation problems simultaneously. Multi-layer group sparsity structure preserves the mutual dependency between the class label and image tags. Secondly, we apply the normalized cut method to form tag-based subgroups. These tag-based subgroups encode the image component information. Based on the predicted class label and the reconstruction error for each tag-based subgroup, we propose a multi-layer group based tag propagation method which improves the robustness of image annotation. Thirdly, we extend our multi-layer group sparse coding in the RHKS and propose kernel multi-layer group sparse coding. Our kernel multi-layer group sparse coding captures the nonlinearity of features, and further improves performances of both image classification and annotation tasks.

This paper is an extension of our previous work [15], and we extend our work in the following aspects: (i) We extend our formulations to more general cases; (ii) Besides F-measure, we also adopt the widely used Precision and Recall for annotation evaluation. (iii) We conduct more experiments for more complete evaluation, like evaluating the effects of different parameters, using a toy dataset to illustrate reason of performing multi-layer group sparse coding in RKHS. (iv) We propose to use k NN strategy to accelerate the proposed kernel multi-layer group sparse coding on large-scale image classification task. (iv) The computational complexity of the solver is analyzed.

The rest of this paper is organized as follows: Related work

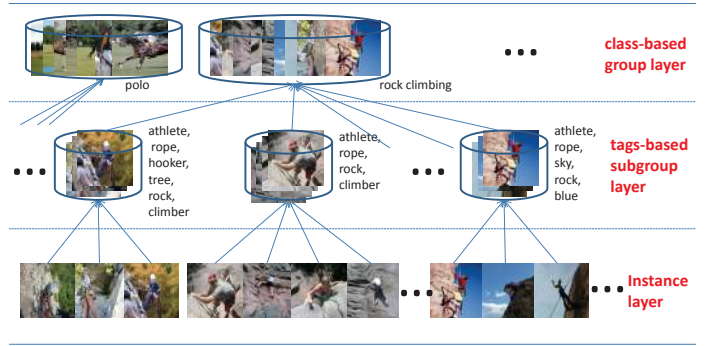


Fig. 2. Multi-layer group structure illustration. The codebook has the multi-layer group structure, therefore the reconstruction coefficients are sparse for different groups in each layer.

will be briefly discussed in Section II. We will describe the details of our multi-layer group sparse coding in Section III. Experiments will be conducted in Section IV. We will conclude our work in Section V.

II. RELATED WORK

Our work is closely related to the image classification, image annotation, and sparse related methods.

Lots of works have been done by using sparse coding for image classification, like sparse coding based spatial pyramid matching [49], Locality-constrained Linear Coding (LLC) [46], Laplacian Sparse coding (LSc) [17], Kernel Sparse Representation (KSR)[16], *etc.* Specifically, given lots of extracted features, like SIFT [30], HOG [11], Marco-feature [5], these sparse coding techniques are used to encode these features. Then max pooling [38] or other feature pooling techniques [6][21] are used to aggregate the information obtained in previous feature coding step for image representation. Moreover, to preserve the spatial information, Spatial Pyramid Matching (SPM) [23] is usually adopted. After representing each image as a vector, Support Vector Machine (SVM) is usually used for training the classifiers and predicting the label of the test image. However, all the above mentioned sparse coding techniques are used for feature coding in image representation process other than used as the classifier for label prediction.

Image annotation methods can be broadly divided into three categories. (i) Generative model [12][32][3]: it propagates key words to the image according to the learnt joint distributions between image features and tags. (ii) Discriminative model [7][10][18][35]: it models the image annotation as a multi-label classification problem and learns a classifier for each key word. (iii) Nearest Neighbor related methods [54][41][19][44]: they propagate the tags of the training samples to the image to be tagged based on their distance/similarity to the training images. In this paper, the proposed technique learns the relationship/similarity between the training images and images to be tagged. Then it propagates the tags of the corresponding training images to the image to be annotated accordingly.

As we aforementioned, image classification and image annotation are closely related for image understanding. Though

jointly solving image classification and annotation is an interesting problem, little research has been done on this topic until now. Recently Wang *et al.* [42] proposed the method of using graphic models (multi-class sLDA and multi-class sLDA with annotations) to tackle such a problem. Moreover, Li *et al.* [25] proposed a method of using a hierarchical generative model to solve image classification, annotation and segmentation. All of these works are based on generative models. However, there are too many parameters in generative models, and their parameter estimation process is usually very computationally expensive.

In our work, we perform the image classification by following the paradigm of sparse coding for face classification [47][53] based on the given image representation, i.e., we predict the label of the test image based on the sparse coefficients. Here sparse coding works as the role of classifier. Moreover, we also model the relationship/similarity between the image to be annotated and all the images used for tag propagation in the same formulation. Mathematically, our multi-layer group sparse coding is closely related to bi-layer sparse coding, group lasso (sometimes it is also called group sparse coding) and sparse group lasso in terms of its formulation but not the application. Compared with sparse coding, bi-layer sparse coding and sparse group lasso encode additional prior information.

Bi-layer sparse coding [27][28] was proposed to tackle label-to-region problem. Bi-layer sparse coding contains the sparsity constraints on two layers: patch-to-patch reconstruction layer and instance layer. Based on its sparse codes, the region to be labeled is connected to several images with different weights. Then related tags of these images can be propagated to each region to be labeled.

Given a test image, the non-zero linear reconstruction coefficients should only appear in one (in the ideal case) or a number of class(es). Thus, group sparse coding lasso was proposed [4]. In group sparse lasso, the reconstruction coefficients corresponding to the instances within the same class form a group, and the ℓ_1 norm is imposed on the group level. It aims at selecting as few classes as possible to reconstruct the test data, and ℓ_2 or ℓ_∞ is usually imposed on the sparse codes within each class. Furthermore, to emphasize the sparse characteristics on both instance layer and group layer, sparse group lasso, which is the combination of group sparse coding and sparse coding, is also introduced to solve regression problems recently [14].

III. MULTI-LAYER GROUP SPARSE CODING

In this section, we will describe the formulation of our multi-layer group sparse coding, its optimization, tag-based subgroup construction, and the prediction of the class label and tags using sparse codes. Whereafter, we will extend our formulation in the RKHS and propose the Kernel Multi-layer Group Sparse coding. Moreover, to handle the large scale problem, we also combine our multi-layer group sparse coding with the k NN, and we also detail the computational cost of such strategy.

A. Problem Formulation

There are tri-layer group sparse constraints in our concurrent image classification and image annotation problem, i.e., instance layer sparse constraint, class-based group layer sparse constraint and tag-based subgroup layer sparse constraint. (i) Given sufficient training images, a test image y can be “sparsely” and linearly reconstructed by the training data of the same class [47]. Here “sparsity” means that only a few reconstruction coefficients are non-zeros. This is the sparsity constraint on the **instance layer**. (ii) To emphasize that the instances used for reconstructing the test data should come from the same class, we gather the sparse codes of the instances within the same class and form a class-based group. We impose the sparsity constraint on these class-based groups: making the groups with non-zero sparse codes as few as possible. This is the sparsity constraint on the **class-based group layer**. (iii) Intuitively, the class label and the tags are closely related. Such co-occurrence information helps the prediction of each other. Thus we introduce an additional layer – **tag-based subgroup layer**, between the instance layer and the class-based group layer. We divide the images within the same class label into different subgroups based on their tag distribution. As shown in Fig. 2, the images within the same subgroup have similar tag distribution, which means the components of these images are similar. Ideally, these subgroups cover all the cases of the tag distribution within each class. Given an instance to be annotated, it can be reconstructed by using the instances within a subgroup under the class-based group that it belongs to. This subgroup contains all the components of an image to be annotated. In practice, because the limitation of the training samples within each class, we cannot get all the subgroups with the same tag distribution. As a compromise, we divide the instances within each class into several subgroups, and the images within each subgroup have similar distribution. We desire that the instances used for reconstruction come from as few subgroups as possible – using the instances from several subgroups are sufficient to reconstruct this test image. Let the reconstruction coefficients corresponding to each subgroup of the training data form a tag-based subgroup. The reconstruction coefficients should be sparse for these tag-based subgroups. Thus the sparsity constraint is introduced on the tag-based subgroup layer. The multi-layer group structure is illustrated in Fig. 2. An illustration of the sparsity on these three layers is depicted in Fig. 3. From Fig. 3, we observe that the reconstruction coefficients, the ℓ_2 norm of the sparse codes on class-based group layer and tag-based subgroups layer are all sparse.

Suppose there are N classes in all. All the training images of the i^{th} class form the matrix X_i ($1 \leq i \leq N$). The images in X_i are divided into G_i tag-based subgroups. Denote the images in the g^{th} tag-based subgroup as X_{ig} ($1 \leq g \leq G_i$), the k^{th} images in the tag-based subgroup X_{ig} as X_{ig}^k ($X_{ig}^k \in \mathbb{R}^{d \times 1}$, $1 \leq k \leq N_{ig}$, here N_{ig} is the number of images in tag-based subgroup X_{ig} , and d is the feature dimension for each image), and X as the matrix of all the images. Then we

have the following relations:

$$\begin{aligned} X_{ig} &= [X_{ig}^1, X_{ig}^2, \dots, X_{ig}^k, \dots, X_{ig}^{N_{ig}}] \in \mathbb{R}^{d \times N_{ig}} \\ X_i &= [X_{i1}, X_{i2}, \dots, X_{ig}, \dots, X_{iG_i}] \in \mathbb{R}^{d \times \sum_g N_{ig}} \quad (1) \\ X &= [X_1, X_2, \dots, X_i, \dots, X_N] \in \mathbb{R}^{d \times \sum_i \sum_g N_{ig}} \end{aligned}$$

Denote the reconstruction coefficient of instance y corresponding to X_{ig}^k , X_{ig} , X_i , X as V_{ig}^k , V_{ig} , V_i and V respectively. They satisfy the following relations:

$$\begin{aligned} V_{ig} &= [V_{ig}^1, V_{ig}^2, \dots, V_{ig}^k, \dots, V_{ig}^{N_{ig}}]^T \in \mathbb{R}^{N_{ig} \times 1} \\ V_i &= [V_{i1}^T, V_{i2}^T, \dots, V_{ig}^T, \dots, V_{iG_i}^T]^T \in \mathbb{R}^{\sum_g N_{ig} \times 1} \quad (2) \\ V &= [V_1^T, V_2^T, \dots, V_i^T, \dots, V_N^T]^T \in \mathbb{R}^{\sum_i \sum_g N_{ig} \times 1} \end{aligned}$$

Based on the previous definition, the sparsity corresponding to the instance layer, class-based group layer and tag-based subgroup layer can be formulated as $\|V\|_1$, $\sum_i \|V_i\|_p$ and $\sum_i \sum_g \|V_{ig}\|_p$. Therefore we formulate the objective of multi-layer group sparse coding as follows:

$$\begin{aligned} \min_V \frac{1}{2} \|y - XV\|_F^2 + \lambda \|V\|_1 + \sum_{i=1}^N w_i \|V_i\|_p \\ + \sum_{i=1}^N w_i \sum_{g=1}^{G_i} \gamma_{ig} \|V_{ig}\|_p \quad (3) \end{aligned}$$

Here λ , w_i and γ_{ig} are the weights on different layers and different groups. ℓ_p norm is used on the sparse codes within each (sub)group. When $p = 1$, the formulation becomes a tri-layer sparse coding, which is closely related to the bi-layer sparse coding [27][28]. However, the correlation of the images within each (sub)group will be lost if ℓ_1 norm is used. In the following sections, we set $p = 2$. Namely ℓ_2 norm is used to encode the sparse codes within each (sub)group as an unit [14][22].

B. Objective Optimization

The objective of our formulation is convex, but it is non-smooth. Moreover, the class-based groups and tag-based subgroups are overlapped with each other. Commonly used methods for solving group sparse coding, like block coordinate descend method [13][31][14][36], are not suitable for optimizing our objective function because of the non-separable variables. Recall that $\|x\|_1 = \sum_i |x_i| = \sum_i \|x_i\|_2$ (where x is a vector, x_i is its i th entry, and $|x_i|$ is absolute value of x_i), therefore we can rewrite Equation (3) as the summation of ℓ_2 norm imposed on each group:

$$\begin{aligned} \min_V \frac{1}{2} \|y - XV\|_F^2 + \lambda \sum_{i=1}^N \sum_{g=1}^{G_i} \sum_{k=1}^{N_{ig}} \|V_{ig}^k\|_2 + \sum_{i=1}^N w_i \|V_i\|_2 \\ + \sum_{i=1}^N w_i \sum_{g=1}^{G_i} \gamma_{ig} \|V_{ig}\|_2 \quad (4) \end{aligned}$$

In this formulation, the groups are in three forms: each instance forms a group (group number is $\sum_{i=1}^N \sum_{g=1}^{G_i} N_{ig}$), class-based group (group number is N), tag-based subgroup (group number is $\sum_{i=1}^N G_i$). The total group number $M = \sum_{i=1}^N \sum_{g=1}^{G_i} N_{ig} + N + \sum_{i=1}^N G_i$. All the groups are indexed

by the set $\mathcal{B} = \{b_1, b_2, \dots, b_M\}$, which is defined as a subset of the powerset of $\{1, 2, \dots, \sum_{i=1}^N \sum_{g=1}^{G_i} N_{ig}\}$. Let the sparse codes and the weight corresponding to group b be V_b and β_b respectively. Then Equation (4) can be further simplified to the following formulation:

$$\min_V \frac{1}{2} \|y - XV\|_F^2 + \sum_{b \in \mathcal{B}} \beta_b \|V_b\|_2 = \frac{1}{2} \|y - XV\|_F^2 + \Omega(V) \quad (5)$$

which is a group lasso problem with overlapped groups. It can also be deemed as a case of tree-structured lasso problem [20]. Recently, Chen *et al.* propose the Proximal-Gradient method [8] to efficiently optimize tree-structured lasso regression problem. In this paper, we adopt this method to optimize Equation (4).

Let $\alpha = [\alpha_{b_1}^T, \alpha_{b_2}^T, \dots, \alpha_{b_M}^T]^T$ ($\alpha \in \mathbb{R}^{\sum_{b \in \mathcal{B}} |b| \times 1}$) be a vector defined on the domain $\mathcal{Q} \equiv \{\alpha \mid \|\alpha_b\|_2 \leq 1, \forall b \in \mathcal{B}\}$. Define C ($C \in \mathbb{R}^{\sum_{b \in \mathcal{B}} |b| \times \sum_{i=1}^N \sum_{g=1}^{G_i} N_{ig}}$) as a matrix, whose rows are indexed by all pairs of $(i, b) \in \{(i, b) \mid i \in b, i \in \{1, 2, \dots, \sum_{i=1}^N \sum_{g=1}^{G_i} N_{ig}\}\}$, and columns are indexed by $j \in \{1, 2, \dots, \sum_{i=1}^N \sum_{g=1}^{G_i} N_{ig}\}$. Each element of C is given as:

$$C_{(i,b),j} = \begin{cases} \beta_b & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Then $f_\mu(V) = \max_{\alpha \in \mathcal{Q}} \alpha^T C V - \frac{\mu}{2} \|\alpha\|_2^2$ is the smooth approximation of $\Omega(V)$. Here μ is a positive smoothness parameter, which is used to control the accuracy of approximation. Smaller μ corresponds to more precise approximation. We set it to 10^{-3} in our experiments. By substituting $\Omega(V)$ with $f_\mu(V)$ in Equation (5), we arrive at the following smooth optimization problem:

$$\min_V \frac{1}{2} \|y - XV\|_F^2 + f_\mu(V) \quad (7)$$

This objective function can be efficiently optimized by using Nesterov's method [34], which is an accelerated gradient method. It can be shown that the solution of Equation (7) can be sufficiently close to the optimal solution V^* of Equation(3).²

C. Tag-Based Subgroup Construction in Multi-layer Group Sparse Coding

One important issue is how to form the subgroups based on the image tag distribution. Each image can be represented by a vector in which each entry is either 1 or 0 representing whether the occurrence of a certain tag in the image or not. However, it may not be proper to adopt traditional k -means to partition the images into clusters because Euclidean distance is not appropriate for evaluating the distance between such tag vectors.

The criteria for our tag-based subgroup construction are given as follows: The inter-group similarity should be as large as possible, and the intra-group similarity should be as small as possible. Therefore, we can formulate such subgroup construction problem as a normalized cut problem [39]. Each

²For more details on proximal-gradient method, please refer to reference [8].

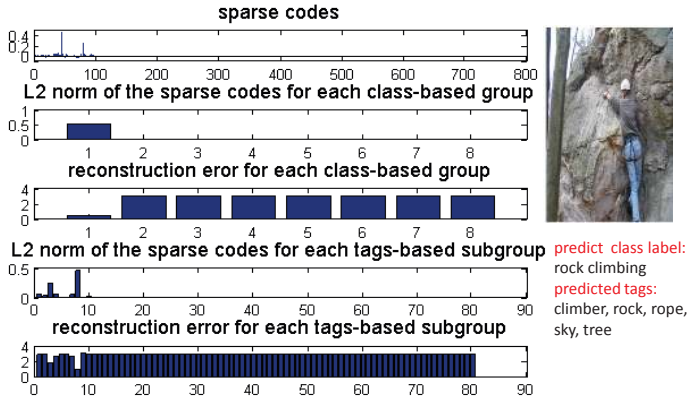


Fig. 3. An illustration of sparse codes, ℓ_2 norm of the sparse codes group on class-based group layer ($\|V_i\|_2$) and tag-based subgroup layer ($\|V_{ig}\|_2$), and reconstruction error of class-based group and tag-based subgroup.

image is a vertex of a graph, the edge between the images is weighted by the similarity between two vertices. By using the normalized cut algorithm, we can cut the whole graph into some subgraphs. The images within each subgraph form a subgroup which satisfies our criteria. Cosine distance is adopted to calculate the similarity between image tag vectors due to its effectiveness in characterizing the similarity between text documents.

D. Class Label and Tag Prediction

Following the previous work [47][52], we predict the test image's class label based on the reconstruction error in the class-based group, and assign the test image to the class with the minimum reconstruction error. The reconstruction error corresponding to the i^{th} class can be computed as $e(i) = \|y - X_i V_i\|_F^2$. The class label of y can be obtained as follows:

$$\text{Class Label of } y \leftarrow \arg \min_i \{e(1), e(2), \dots, e(i), \dots, e(N)\} \quad (8)$$

After predicting the class label of the instance, we select some tag-based subgroup(s) of the predicted class to annotate the test image. The sparsity of sparse codes on the three layers and the reconstruction error for each class-based group and tag-based subgroup are shown in Fig. 3. The (sub)groups with non-zero ℓ_2 norm of the sparse codes and smaller reconstruction error are closely related to the test image. To be consistent with previous work, the selection of subgroup(s) is(are) also based on the reconstruction error. Suppose the predicted class label of the test image y is i . Then we can calculate the reconstruction error corresponding to each tag subgroup X_{ig} : $r(g) = \|y - X_{ig} V_{ig}\|_F^2$ ($1 \leq g \leq G_i$), and sort the reconstruction error for these subgroups in ascending order. The resultant subgroup index order after such sorting is $\{g_1, g_2, \dots, g_j, \dots\}$, which is a permutation of $\{1, 2, \dots, G_i\}$. Suppose the tag matrix for subgroup g_i is T_{g_i} , in which each column is the tag representation of one image. We first use the tags in subgroup g_1 to annotate the test image. To annotate the image with k tags, we weight the tags within subgroup g_1 by the sparse codes ($T_{g_1} V_{g_1}$). Then we sort these tags in subgroup

Algorithm 1 Multi-layer group Based Tag Propagation

- 1: **INPUT:** The test data y and its predicted class label i ; Training data: X_{ig} ; Sparse codes: V_{ig} ; Tag matrix: T_{ig} , $g \in \{1, 2, \dots, G_i\}$; Tag number of image y : k ;
- 2: **OUTPUT:** The tag set for y : $T(y)$.
- 3: Calculate reconstruction error for each subgroup: $r(g) = \|y - X_{ig} V_{ig}\|_F^2$;
- 4: Sort $r(g)$ in ascending order and get the corresponding group index: $[g_1, g_2, \dots, g_j, \dots, g_{G_i}]$.
- 5: Initialize $j = 1$, $T(y) = \emptyset$;
- 6: **WHILE** ($|T(y)| < k$ && $j \leq G_i$)
- 7: Weight tags using sparse codes: $T_{tmp} = T_{ig_j} V_{ig_j}$;
- 8: Sort T_{tmp} ;
- 9: Propagate top ($k - |T(y)|$) tags which are not included in $T(y)$ to $T(y)$ according to T_{tmp} ;
- 10: $j = j + 1$;
- 11: **END**

g_1 and propagate the top k tags to the test image. If the tag number in subgroup g_1 is less than k , then we will sequentially use subgroup g_2 , g_3 , and so on, until the tag number is reached. The details of such multi-layer group based tag propagation method are given in Algorithm 1.

There are some advantages for our multi-layer group based tag propagation method. First of all, using all the images within each subgroup enhances the robustness of image annotation. Though those images with higher weight are more important, the test image may not have exactly the tag distribution with the image with the largest sparse code. So it may not be stable to use only one image for tag annotation. To overcome such a problem, we use all the images within the subgroup for tag propagation. Secondly, those images with higher weights play more important roles for the reconstruction of the test image. Weighting the instances within each subgroup with their corresponding sparse codes can emphasize the different importance of different instance for the reconstruction of the test image. Please refer to Section IV-F for the comparisons between different tag propagation methods.

E. Multi-layer Group Sparse Coding in RKHS

Recently, kernel methods [37] have been successfully applied to sparse coding problems [16][53], and experimental results show that the kernel sparse representation can achieve higher accuracy for both image classification and face recognition. Moreover, the sparse codes learnt by the kernel sparse representation can capture nonlinear similarity between the instances, and is more discriminative than that of learnt by general sparse coding.

Motivated by the excellent properties of kernel methods, we propose the kernel multi-layer group sparse coding, which is the multi-layer group sparse coding in a high dimensional space mapped by some explicit function $\phi(\cdot)$. The objective

of kernel multi-layer group sparse coding is given as follows:

$$\min_V \frac{1}{2} \|\phi(y) - \phi(X)V\|_F^2 + \lambda \|V\|_1 + \sum_{i=1}^N w_i \|V_i\|_2 + \sum_{i=1}^N w_i \sum_{g=1}^{G_i} \gamma_{ig} \|V_{ig}\|_2 \quad (9)$$

By using the kernel trick: $\phi(x)^T \phi(y) = \kappa(x, y)$, we can rewrite Equation (9) as follows:

$$\min_V \frac{1}{2} (\kappa(y, y) - 2V^T K_X(y) + V^T K_{XX} V) + \lambda \|V\|_1 + \sum_{i=1}^N w_i \|V_i\|_2 + \sum_{i=1}^N w_i \sum_{g=1}^{G_i} \gamma_{ig} \|V_{ig}\|_2 \quad (10)$$

where K_{XX} is a $\sum_{i,g} N_{ig} \times \sum_{i,g} N_{ig}$ matrix with $\{K_{XX}\}_{ij} = \kappa(x_i, x_j)$, and $K_X(y)$ is a $\sum_{i,g} N_{ig} \times 1$ vector with $\{K_X(y)\}_i = \kappa(x_i, y)$. We also adopt the proximal-gradient method to optimize this objective. The computational cost of kernel multi-layer group sparse coding is the same as that of multi-layer group sparse coding except for the additional cost for pre-computing the kernel matrix.

As we use spatial pyramid representation for each image, which is a combination of visual word histograms in different spatial regions, we will use Histogram Intersection Kernel (HIK) due to its excellent performance in evaluating the similarity between two histograms [48]. Another advantage for choosing HIK is that HIK is a parameter-free kernel. The HIK between two normalized histograms Y_1 and Y_2 is defined as $K(Y_1, Y_2) = \sum_i \min(Y_{1i}, Y_{2i})$.

F. Accelerating Multi-layer Group Sparse Coding with k NN

In our method, the computational cost usually increases with the size of the dictionary [24], which restricts the proposed formulation for the classification for datasets with many categories. To solve this problem, we propose to combine our method with k NN strategy. That is, we select only a few categories based on the similarity of the test image to the training categories, and the image to category similarity is calculated as the average similarity of the test sample to all the training sample of that category. Then only the top k categories are selected and used as the refined dictionary. Another reason for incorporating k NN into our multi-layer group sparse coding comes from the work of Locality-constrained Linear Coding [46] which uses the k NN strategies to refine the codebook first for the feature to be encoded. In this way, the locality information of the feature to be encoded can be preserved and usually improves the feature coding quality.

Specifically, the main computational cost of our method comes from the optimization of Equation (7). According to [8], to achieve the ϵ accuracy for the optimization of Equation (7), the total computational cost is $O(J^2 d + (J^2 + \sum_{b \in \mathcal{B}} |b|)/\epsilon)$ (Here $J = \sum_i \sum_g N_{ig}$ is the size of the codebook X , and $|\mathcal{B}| = \sum_i \sum_g N_{ig} + N + \sum_i G_i$ is the number of the groups). After using k NN strategy, suppose the indices corresponding to the selected k categories form a set S , i.e., $|S| = k$, then the codebook size to $J_{kNN} = \sum_{i \in S} \sum_g N_{ig}$, and the

number of the groups corresponding to the reduced codebook becomes $|\mathcal{B}_{kNN}| = \sum_{i \in S} \sum_g N_{ig} + k + \sum_{i \in S} G_i$. As a result, by using the k NN strategy, the total computational cost for achieving ϵ accuracy for objective (7) becomes $O(J_{kNN}^2 d + (J_{kNN}^2 + \sum_{b \in \mathcal{B}_{kNN}} |b|)/\epsilon)$. Compared with the computational cost without using k NN, for large-scale dataset, $k \ll N$, as a result, $J_{kNN} \ll J$ and $|\mathcal{B}_{kNN}| \ll |\mathcal{B}|$. Take NUS-WIDE-Object as an example, there are 26 categories, and the time for calculating the KMIGSC coefficients is around 10.87 second, but using k NN ($k=10$), the computational cost greatly reduced to 2.35 second, which is about 4.6 times faster than the original algorithm, but the performance is still comparable (For more details, please refer to Section IV). Therefore we can conclude that k NN strategy can greatly reduce the total computational cost and make our algorithm possible to handle large-scale dataset.

G. Discussions

In the previous sections, the formulations are proposed based on the application in concurrent image classification and annotation, and only three layers are considered. We can generate our formulation to more general cases. Denote the reconstruction coefficients corresponding to the g^{th} group as V_g , and denote the norm on V_g as $\|V_g\|_p$ ($p = 2$ or \inf), then we can get the following more general multi-layer group sparse coding problem (Equation 11). We can also extend this more general multi-layer group sparse coding to RKHS, and get the Kernel Multi-layer group sparse formulation in Equation 12.

$$\min_V \frac{1}{2} \|y - XV\|_F^2 + \sum_g \lambda_g \|V_g\|_p \quad (11)$$

$$\min_V \frac{1}{2} \|\phi(y) - \phi(X)V\|_F^2 + \sum_g \lambda_g \|V_g\|_p \quad (12)$$

As shown in Fig. 4, based on the group structure (overlapped, non-overlapped, or tree-structure), we can get overlapping group lasso, non-overlapping group lasso and tree-guided group lasso, and similar objective optimization strategy can be used.³ In real applications, we can select the corresponding formulation based on the specific task we are tackling.

IV. EXPERIMENTS

In this section, we will experimentally evaluate the proposed method for image classification and annotation as well as the effect of different parameters.

A. Classification of Sparse Representation in RKHS on Toy Data

Previous work has shown the effectiveness of bi-layer sparse coding [14][29], so here we propose to use the toy data to demonstrate the reason of using sparse coding in RKHS, i.e., Kernel Sparse Representation (KSR) [16] for the

³For more details about the optimization of different variants, please refer to [8].

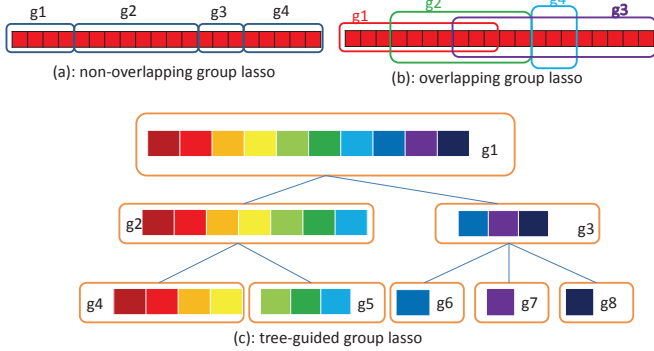


Fig. 4. An illustration of different formulations under different group structure.

classification of nonlinear separable data. Though KSR has been proposed by Gao *et al.* [16], none experimentally evaluate it on simulated data. To illustrate its usefulness for nonlinearly separable data, we generate two classes as follows: The data collection C1 contains 1000 instances and each instance is 30D. Each dimension is uniform distributed among $[0, 1]$, then we normalize the ℓ_2 norm of each instance to 1. So the data in C1 are distributed on the ball with radius 1. The data collection C2 contains 1000 instances and each instance is 30D. Each dimension is uniform distributed among $[0, 1]$, then we normalize the ℓ_2 norm of each instance to 1.2. So the data in C2 are distributed on the ball with radius 1.2. We illustrate the data used in the experiments in Fig. 5. In this experiment, we sample 800 instances from C1 and C2 randomly as training samples, and use the rest (400) as test data to perform the classification problem by following the paradigm of sparse coding for face recognition [47], i.e., we predict the label of the instance according to the reconstruction error. For Kernel Sparse representation, we use the Gaussian kernel ($\kappa(x_1, x_2) = \exp(-\gamma\|x_1 - x_2\|^2)$) and set the parameter γ in Gaussian kernel to be $1/d$ (d is the dimension of the data. Here it is 30.). We list the performance of sparse coding and KSR in table I. We can see that our Kernel Sparse representation greatly outperforms the sparse coding for the nonlinearly separable data. For our image classification in this paper, the SPM or BoW model is chosen for image representation, which is also nonlinearly separable, therefore it convinces us we can perform the kernel multi-layer group sparse coding to further boost the performance of image classification.

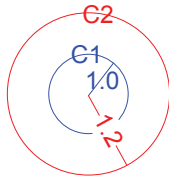


Fig. 5. An illustration of toy data used in simulated experiments. Data from C1 are distributed on the blue ball, and data from C2 are distributed on the red ball.

4

TABLE I
PERFORMANCE OF SPARSE CODING IN RKHS ON TOY DATA. THE VALUE IN THE BRACKET IS THE THE WEIGHT OF SPARSITY TERM.

	Sc(0.01)	KSR(0.01)	Sc(0.05)	KSR(0.05)
accuracy (%)	49.60±0.46	83.50±0.89	50.08±0.24	83.10±2.0
sparsity (%)	0.98	1.02	0.54	0.97

B. Dataset Description

We use the following datasets to evaluate our methods: UIUC-Sport and LabelMe datasets which are used in the work of Wang *et al.* [42], and NUS-WIDE-Object [9] dataset. UIUC-Sport dataset contains 1792 images which are classified into 8 classes. We resize the max side (width or height) of images to 400 pixels and keep the aspect ratio. Following the setting of Wang *et al.* [42], we evenly split the data into training and test data for each class. LabelMe dataset [42] also contains 8 classes. To be consistent with the work of Wang *et al.* [42], we only use images with 256×256 pixels. We randomly select 100 images as training data and randomly select another 100 images as test data. The total image number is 1600. NUS-WIDE-Object dataset contains 30,000 images and 31 classes. We use 26 classes in our experiments,⁵ and 120 images are used as the training data and 40 images as the test data for each class. Need to mention that we only use the images with single class label and multiple tags. The number of image tags on the LabelMe, UIUC-Sport and NUS-WIDE-Objects databases are 274, 175 and 813 respectively.

C. Experimental Setup

Following the work of Wang *et al.* [42], we also adopt densely-sampled SIFT feature, whose step size and patch size are 4 and 24 respectively. Then we quantize all the features into 400 clusters by using k -means. To preserve the spatial information, spatial pyramid representation [23] is also used.⁶ We use three levels, and the weights corresponding to different levels are all 1. ℓ_2 norm is used to normalize the histogram inside each level. For image annotation, we get rid of the tags whose frequencies are less than three. For the number of tag-based subgroups, we set $G_i = 10$ for each class. We repeat the experiments 10 times independently on the LabelMe and UIUC-Sport databases, and we use the first 120 (40) images that have single class label and multiple tags in the standard partition of training (test) set provided by the NUS-Wide-Object database.

Following the work of Wang *et al.* [42], we use the classification accuracy to evaluate the performance of image classification, and use F-measure of top K tags to evaluate the performance of image annotation. To be consistent with the work of [42], we also set $K = 5$ in our experiments. Moreover, we also report the performances of different methods under the measurement of Precision and Recall, which are commonly

⁵*books, flags, zebra, computer and whales* are not used due to the very small number of training images.

⁶We use BoW features for the NUS-WIDE-Object dataset because spatial pyramid representation is not provided by this dataset, and BoW is a histogram feature. HIK is also suitable for BoW features.

TABLE II

PERFORMANCE COMPARISON BETWEEN UNIFORM SETTING (UNIFORM) AND GROUP SIZE WEIGHTED PARAMETER SETTING (WEIGHTED) (%).

setting	LabelMe		UIUC-Sport	
	Accuracy	F measure	Accuracy	F measure
Uniform	75.69	42.27	76.78	52.04
Weighted	76.24	43.43	76.63	53.25

used criteria for image annotation evaluation. Define Precision as the percentage of correct tags out of the top K propagated tags, and define Recall as the percentages of correct tags propagated out of all the ground-truth tags of the image. Then F-measure is defined as follows:

$$F\text{-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (13)$$

D. Parameter Evaluation

To reduce the amount of parameter tunings in (kernel) multi-layer group sparse coding, we use the following two methods to simplify the parameters setting:

Uniform Setting. We set $w_i = w_0, \forall i \in [1, N], w_i \times \gamma_{ig} = \gamma_0, \forall i \in [1, N], g \in [1, G_i]$. In this way, there are 3 parameters in Equation (3) and Equation (9): λ, w_0 and γ_0 . In our experiments, we set $w_0 = \gamma_0 = 10^{-3}, \lambda = 10^{-2}$.

Group Size Weighted Parameter Setting. Such weighting strategy is designed for group lasso, and it is used by Chen *et al.* [8] and Yuan *et al.* [51]. The weight for certain group is proportional to the square root of the instance number in that group. That is: $\beta_b = \theta \sqrt{|V_b|}$ in Equation (5). Then we can get $\lambda = \theta, w_i = \theta \sqrt{|V_i|}$ and $w_i \times \gamma_{ig} = \theta \sqrt{|V_{ig}|}$ in Equation (3) and Equation (9). In this way, there is only one parameter θ in the whole formulation. In our experiments, we set $\theta = 10^{-3}$.

We list the performance of these two parameter setting methods on the UIUC-Sport and LabelMe datasets in Table II. From the results, we can see that the group size weighted parameter setting usually achieves better performance than the uniform setting. More importantly, there is only one parameter in such parameter setting method, which can greatly simplify the parameters in the formulations of our method. Thus such group size weighted parameter setting is adopted in the following experiments. Need to mention that there are also some other parameter setting methods, for example, combining uniform setting and group size weighted parameter setting. Our method may achieve even better performance by using some other advanced parameter setting methods.

E. Performance Comparison

We compare our multi-layer group sparse coding (MIGSc) and kernel multi-layer group sparse coding (KMIGSc) with the following related work: (i): multi-class sLDA with annotations (McsLDAA) [42]. (ii): Sparse coding with instance based annotation (ScIBA). For image classification, we use sparse coding framework [47]. For annotation, we adopt the strategy of instance based annotation (please refer to Section IV-F for details). (iii): Sparse group lasso and instance based annotation (SGLIBA). Sparse group lasso corresponds to the bi-layer group sparse coding with sparsity on instance layer

TABLE III

PERFORMANCE COMPARISONS BETWEEN DIFFERENT METHODS FOR CONCURRENT IMAGE CLASSIFICATION AND ANNOTATION (%).

	LabelMe			
	Accuracy	F-measure	Precision	Recall
McsLDAA [42]	76.0	38.7	NA	NA
ScIBA [47]	75.24	31.21	45.97	38.22
SGLIBA [14]	75.13	37.24	43.29	36.18
MIGSC	76.24	43.43	49.94	42.53
KMIGSC	82.94	48.12	55.21	47.23
	UIUC-Sport			
	Accuracy	F-measure	Precision	Recall
McsLDAA [42]	66.0	35.0	NA	NA
ScIBA [47]	74.32	48.02	57.66	43.26
SGLIBA [14]	76.15	44.21	52.92	39.93
MIGSC	76.63	53.25	63.54	48.20
KMIGSC	79.37	55.43	66.11	50.15
	NUS-WIDE-Object			
	Accuracy	F-measure	Precision	Recall
ScIBA [47]	18.85	6.01	6.44	6.19
SGLIBA [14]	18.65	7.9	6.43	11.75
MIGSC	19.52	9.74	6.22	22.22
KMIGSC	20.96	10.48	11.38	10.81

and class-based group layer. We use sparse group lasso [14] for image classification, and use instance based annotation method to annotate the test image. The performance comparisons between different methods are given in Table III.⁷

Table III shows that our methods outperform the baseline methods in both image classification and annotation tasks on all the datasets. We can see that our multi-layer group sparse coding outperforms sparse coding and sparse group lasso, which shows the effectiveness of our tag-based subgroup layer. By using the kernel technique, our method achieves the best performance. Compared with multi-class sLDA with annotations, the performance of our method increases by more than 10% for image classification, and our method also improves the F-measure by 6.94% and 13.37% respectively for image annotation on the LabelMe and UIUC-Sports datasets. These demonstrate the effectiveness of our multi-layer group structure.

We also list the confusion matrices of the LabelMe and UIUC-Sport datasets in Fig. 6. From the results, we can observe that some classes are easily misclassified to each other, such as “*croquet*” and “*bocce*”, “*street*” and “*inside city*”, *etc.* We list some classification and annotation results of our method in Fig. 7.

F. Comparison Between Different Annotation Strategies

We compare our multi-layer group based tag propagation method with the following methods. The class label information is not used in the following methods.

Instance based annotation. Since the magnitude of the sparse codes hints at the importance of certain training images in the reconstruction of the test image, the priority of the training images used for tag propagation is based on the magnitude of its corresponding sparse code. Then, the tag propagation priority for each image is based on their frequencies in the whole dataset.

⁷We set $\theta = 5 \times 10^{-3}$ in group size weighted parameter setting on the NUS-WIDE-Object dataset because of different features.












			
rock climbing climber, rock, rope, hook, knapsack	sailing athlete, sky, hill, sailing boat, water	rowing athlete, floater, oar, plant, rowboat	rowing athlete, battledore, floor, net, wall
rock climbing climber, rock, rope, hook, knapsack, plant, sky	sailing athlete, sky, hill, sailing boat, water	rowing athlete, floater, oar, plant, rowboat	rowing athlete, battledore, floor, net, wall, door,
			
snowboarding house, plant, ski, skier, sky	croquet athlete, grass, mallet, mallet, wicket	polo athlete, grass, horse, mallet, tree	bocce ball, lawn, sky, stand, tree
snowboarding house, plant, snow field, skier, sky, audience, ski	croquet athlete, grass, mallet, plant croquet, wicket, ground	polo athlete, grass, horse, mallet, tree, ball	bocce ball, lawn, sky, stand, tree, athlete, coach, drink
			
bocce athlete, ball, lawn, sky, tree	sailing athlete, boat, building, sky, sailing boat	croquet athlete, ball, lawn, mallet, tree	croquet athlete, ball, lawn, mallet, tree
croquet athlete, ball, lawn, sky, tree, crosspiece, mallet, wicket	rowing athlete, boat, building, sky, bank, oar, lake, tree	bocce athlete, ball, lawn, grass, tree, coach, spectator	polo athlete, ball, lawn, mallet, tree, sign, horse

Fig. 7. Some results on the UIUC-Sport dataset. The class label/tags under the images (in black) are the results of our method. The ones in red are wrongly predicted, and the ones in blue are ground-truth. We also list some images with easily-misclassified class label in the last row.

Subgroup based annotation without the inference of class label. We rank all the tag-based subgroups in ascending order according to their reconstruction error. The subgroup with the minimum reconstruction error is firstly used as the unit for tag propagation. We also weight the tag matrix using the sparse codes and propagate those tags with higher weight first.

Greedy label transfer [1][2]. This method annotates the test image based on its k nearest neighbors, tag frequency and tag co-occurrence information.

We evaluate the capability of these methods for image annotation on the LabelMe and UIUC-Sport databases. Besides F-measure, we also adopt Precision and Recall as other evaluation criteria. Fig. 8 shows the good performance of our multi-layer group based image annotation in terms of all the evaluation criteria, and it verifies the effectiveness of the class label, and the robustness of subgroup based methods in image annotation.

G. The Effect of The Number of Subgroups

In our work, graph cut is used to determine the tag-based subgroups, and it performs the role of clustering. The determination of the group number in clustering is an open problem. Therefore it is not easy to determine the best number of the subgroups. In our experiments, because the images in each category are not too many (around 100 images), for simplification, we fix this parameter to a small value (10) on all the datasets, and each subgroup has about 10 images. If we have enough samples for each category, we then can increase this parameter to better characterize the distribution of the tags. Specifically, we test the performance of our algorithm on UIUC-Sports by setting it to be 5, 8, 10, 12. We show the performances under different parameter setting in Fig. 9. We can see that the small change of the group number has little effect on the final performance. Please note that if we further increase this number to the the images numbers within each class (the largest value it can be, around 100 for UIUC-Sports and LabelMe), our multi-layer group sparse coding corresponds to the case of sparse group lasso, which is a

	coast	forest	highway	inside city	mountain	open country	street	tall building
coast	73.0%	0.5%	9.2%	0.1%	2.7%	12.0%	1.8%	0.7%
forest	0.1%	94.3%	0.4%	0.0%	3.1%	1.4%	0.7%	0.0%
highway	7.2%	1.0%	80.7%	0.5%	1.9%	5.3%	1.8%	1.6%
inside city	2.5%	0.5%	4.0%	76.9%	0.2%	1.3%	8.6%	6.0%
mountain	5.5%	2.5%	6.5%	0.3%	70.9%	8.2%	3.3%	2.8%
open country	18.9%	7.2%	6.3%	0.2%	6.3%	57.0%	3.0%	1.1%
street	1.3%	0.2%	5.1%	12.9%	1.7%	1.6%	74.9%	2.3%
tall building	1.7%	2.5%	4.8%	5.2%	2.9%	1.7%	2.2%	79.0%

Confusion Matrix on the LabelMe dataset

	rockclimbing	badminton	bocce	croquet	polo	rowing	sailing	snowboarding
rockclimbing	92.9%	0.1%	0.7%	0.1%	0.4%	0.3%	3.1%	2.4%
badminton	1.3%	89.8%	1.1%	0.7%	0.5%	1.6%	1.9%	3.1%
bocce	4.9%	7.0%	48.7%	25.7%	2.9%	3.5%	3.5%	3.9%
croquet	3.1%	0.8%	9.2%	80.6%	1.7%	1.9%	1.9%	0.8%
polo	0.4%	2.3%	1.6%	8.2%	78.8%	2.2%	3.8%	2.5%
rowing	1.5%	1.3%	1.3%	1.9%	1.2%	86.7%	5.0%	1.1%
sailing	2.1%	0.6%	0.2%	1.8%	0.5%	11.1%	76.1%	7.6%
snowboarding	10.9%	2.3%	2.3%	2.1%	1.9%	5.9%	15.1%	59.5%

Confusion Matrix on the UIUC-Sport dataset

Fig. 6. Confusion matrices of our multi-layer group sparse coding on the LabelMe and UIUC-Sport datasets.

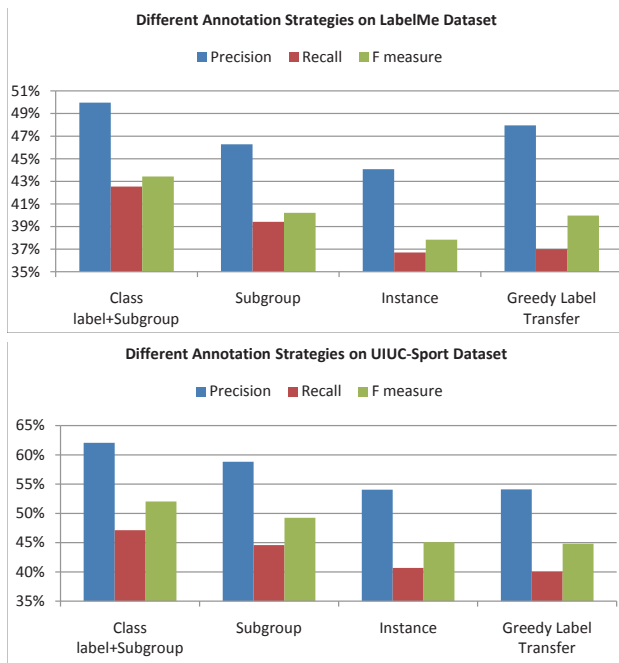


Fig. 8. Performance comparison based on different annotation strategies on the LabelMe and UIUC-Sport datasets.

combination of sparse coding and group lasso. Table III shows that our method outperforms sparse group lasso by 1% in terms of classification accuracy on all datasets, and 6.3%, 9.0%, and 1.8% in terms of F-measure on LabelMe, UIUC-Sport and NUS-WIDE-Object dataset. This also proves the usefulness of our multi-layer group sparse coding framework.

The Effect of the Number of Subgroups

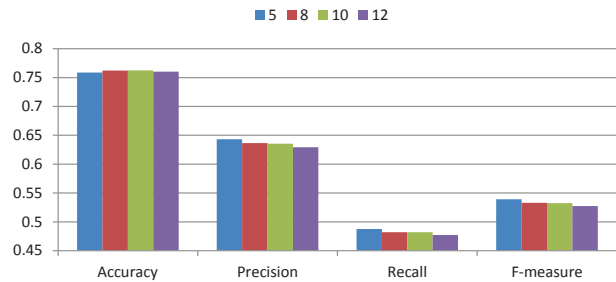


Fig. 9. The effect of varying the number of subgroups on UIUC-Sports.

H. Acceleration with Refined Dictionary

Here we evaluate the performance of Multi-layer group sparse coding with k NN technique on NUS-WIDE-Object dataset. Because BoW representation is used, we use the histogram intersection to evaluate the similarity between two images. The results of the method are listed in Table IV. We can see that by using k NN strategy the computational cost can be greatly reduced but the performance is still comparable with that without using k NN.

TABLE IV
CLASSIFICATION PERFORMANCE BY USING k NN. THE NUMBER IN THE BRACKET IS k .

	Accuracy (%)	time (sec)
KMIGSC	20.96	10.87±1.78
KMIGSC(10)	19.52	2.35±0.32
KMIGSC(15)	20.19	4.92±0.83

V. CONCLUSION

This paper presents a multi-layer group sparse coding framework for concurrent image classification and annotation problems. The main contribution of our method is our multi-layer group sparse structure, which encodes the class information, tag distribution information as well as the mutual dependency between them. Based on such information, we propose the multi-layer group based tag propagation method to annotate the test image. We also extend our work in the RKHS and propose kernel multi-layer group sparse coding. Furthermore, we also integrate our method with the k NN strategy, which greatly improves the computational efficiency. Experimental results show that our method can achieve excellent performances in both image classification and annotation tasks.

It is worth noting that this paper focuses on the study of concurrent single-label image classification and image annotation; however, recently there is a multi-label image classification task [33] on, for example, PASCAL VOC datasets, where multiple objects appear in the same image. Following [45], one future direction is to apply sparse coding based method to solve multi-label image classification problem.

VI. ACKNOWLEDGEMENT

This study is partially supported by the research grant for the Human Sixth Sense Programme at the Advanced

Digital Sciences Center from Singapore's Agency for Science, Technology and Research (A*STAR).

REFERENCES

- [1] M. Ameesh, P. Vladimir, and K. Sanjiv. A new baseline for image annotation. In *Proceedings of the European Conference on Computer Vision*, 2008.
- [2] M. Ameesh, P. Vladimir, and K. Sanjiv. A new baseline for image annotation. *International Journal of Computer Vision*, 2010.
- [3] K. Barnard, P. Duygulu, D. Forsyth, N. D. Freitas, D. M. Blei, T. H. J. K. T. Poggio, and J. Shawe-taylor. Matching words and pictures. *Journal of Machine Learning Research*, 21(3):1107C1135, 2003.
- [4] S. Bengio, F. Pereira, Y. Singer, and D. Strelow. Group sparse coding. In *Proceedings of the Conference on Neural Information Processing Systems*, 2009.
- [5] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [6] Y.-L. Boureau, N. L. Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the locals: multi-way local pooling for image recognition. In *Proceedings of IEEE Conference on Computer Vision*, 2011.
- [7] G. Carneiro, A. Chan, P. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *PAMI*, 29(3):394–410, 2007.
- [8] X. Chen, Q. Lin, S. Kim, J. Peña, J. G. Carbonell, and E. P. Xing. An efficient proximal-gradient method for single and multi-task regression with structured sparsity. Technical report, arXiv:1005.4717, 2010.
- [9] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng. NUS-WIDE: A real-world web image database from National University of Singapore. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, 2009.
- [10] C. Cusano, G. Ciocca, and R. Schettini. Image annotation using svm. In *SPIE*, 2004.
- [11] N. Dalal and W. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2005.
- [12] S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2004.
- [13] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1), 2010.
- [14] J. Friedman, T. Hastie, and R. Tibshirani. A note on the group lasso and a sparse group lasso. Technical report, arXiv:1001.0736v1, 2010.
- [15] S. Gao, L.-T. Chia, and I. W.-H. Tsang. Multi-layer group sparse coding for concurrent image classification and annotation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [16] S. Gao, I. W. Tsang, and L.-T. Chia. Kernel sparse representation for image classification and face recognition. In *Proceedings of the European Conference on Computer Vision*, 2010.
- [17] S. Gao, I. W.-H. Tsang, and L.-T. Chia. Laplacian sparse coding, hypergraph Laplacian sparse coding, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- [18] D. Grangier and S. Bengio. A discriminative kernel-based model to rank images from text queries. *PAMI*, 30(8):1371–1384, 2008.
- [19] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Proceedings of IEEE Conference on Computer Vision*, 2009.
- [20] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. In *Proceedings of the International Conference on Machine Learning*, 2010.
- [21] Y. Jia, C. Huang, and T. Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.
- [22] S. Kim and E. P. Xing. Tree-guided group lasso for multi-task regression with structured sparsity. In *Proceedings of the International Conference on Machine Learning*, 2010.
- [23] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.
- [24] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Proceedings of the Conference on Neural Information Processing Systems*, 2006.
- [25] L.-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding: classification, annotation and segmentation in an automatic framework. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [26] X. Liu, B. Cheng, S. Yan, J. Tang, T.-S. Chua, and H. Jin. Label to region by bi-layer sparsity priors. In *Proceedings of the ACM international conference on Multimedia*, 2009.
- [27] X. Liu, B. Cheng, S. Yan, J. Tang, T.-S. Chua, and H. Jin. Label to region by bi-layer sparsity priors. In *Proceedings of the ACM international conference on Multimedia*, 2009.
- [28] X. Liu, S. Yan, J. Luo, J. Tang, Z. Huang, and H. Jin. Nonparametric label-to-region by search. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [29] X. Liu, S. Yan, J. Luo, J. Tang, Z. Huang, and H. Jin. Nonparametric label-to-region by search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [30] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision(IJCV)*, 60(2):91–110, 2004.
- [31] L. Meier, S. van de Geer, and P. Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B*, 70:53–71, 2008.
- [32] F. Monay and D. Gatica-Perez. PLSA-based image autoannotation: Constraining the latent space. In *ACM Multimedia*, 2004.
- [33] G. Nasierding, G. Tsoumakas, and A. Kouzani. Clustering based multi-label classification for image annotation and retrieval. In *IEEE International Conference on Systems, Man and Cybernetics*, 2009.
- [34] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005.
- [35] J. Petterson and T. Caetano. Reverse multi-label learning. In *NIPS*, 2010.
- [36] K. Rosenblum, L. Zelnik-Manor, and Y. C. Eldar. Dictionary optimization for block-sparse representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2010.
- [37] B. Schölkopf, A. J. Smola, and K.-R. Müller. Kernel principal component analysis. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 583–588, 1997.
- [38] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.
- [39] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
- [40] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, 2003.
- [41] J. Tang, Z.-J. Zha, D. Tao, and T.-S. Chua. Semantic-gap oriented active learning for multi-label image annotation. *IEEE Transactions on Image Processing*, 21(4), 2012.
- [42] C. Wang, D. Blei, and L. Fei-Fei. Simultaneous image classification and annotation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [43] C. Wang, S. Yan, and H.-J. Zhang. Large scale natural image classification by sparsity exploration. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2009.
- [44] C. Wang, S. Yan, L. Zhang, and H. Jiang Zhang. Multi-label sparse coding for automatic image annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [45] C. Wang, S. Yan, L. Zhang, and H.-J. Zhang. Multi-label sparse coding for automatic image annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [46] J. Wang, J. Yang, K. Yu, F. Lv, and Y. Gong. Locality-constrained linear coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [47] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, 2009.
- [48] J. Wu and J. M. Rehg. Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. In *Proceedings of the IEEE International Conference on Computer Vision*, 2009.
- [49] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [50] M. Yang, L. Zhang, J. Yang, and D. Zhang. Robust sparse coding for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

- [51] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B*, 68:49–67, 2006.
- [52] X. Yuan and S. Yan. Visual classification with multi-task joint sparse representation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [53] X.-T. Yuan and S. Yan. Visual classification with multi-task joint sparse representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [54] S. Zhang, J. Huang, Y. Huang, and D. Metaxas. Automatic image annotation using group sparsity. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
- [55] S. Zhang, J. Huang, Y. Huang, Y. Yu, H. Li, and M. Dimitris. Automatic image annotation using group sparsity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.



Zhixiang Ren received the B.E. degree in computer science and technology from the Xidian University in 2006, and M.S.E. degree from Chinese Academy of Sciences in 2009. She is currently pursuing the Ph.D. degree at the School of Computer Engineering, Nanyang Technological University. Her research interests include multimedia signal processing, content/perceptual-based video analytics, computer vision, machine learning etc.

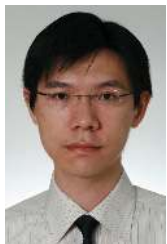


Shenghua Gao received the Ph.D. degree from Nanyang Technological University, Singapore in 2013, and received B.E. degree from the University of Science and Technology of China in 2008. He was awarded the Microsoft Research Fellowship in 2010. He is currently a Postdoctoral researcher in Advanced Digital Sciences Center, Singapore. His research interests include machine learning and computer vision.



Liang-Tien Chia received the B.S. and Ph.D. degrees from Loughborough University (of Technology), Loughborough, U.K., in 1990 and 1994, respectively. He is an Associate Professor in the School of Computer Engineering, Nanyang Technological University, Singapore. He was the Director, Centre for Multimedia and Network Communications from 2002-2007 and Head, Division of Computer Communications during 2007-2010. His research interests can be broadly categorized into the following areas: Internet-related research with emphasis on the

semantic web; multimedia understanding through media analysis, annotation, and adaptation; multimodal data fusion; and multimodality ontology for multimedia. He has published over 100 refereed research papers.



Ivor W. Tsang received the Ph.D. degree in computer science from the Hong Kong University of Science and Technology in 2007. He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. He is also the Deputy Director of the Center for Computational Intelligence, NTU. Dr. Tsang received the IEEE Transactions on Neural Networks Outstanding 2004 Paper Award in 2006, and the second class prize of the National Natural Science Award 2008, China in 2009. His works

earned him the Best Paper Award at ICTAI'11, the Best Student Paper Award at CVPR'10, and the Best Paper Award from the IEEE Hong Kong Chapter of Signal Processing Postgraduate Forum in 2006. He was also conferred with the Microsoft Fellowship in 2005.