International Journal of Computer Vision
https://doi.org/10.1007/s11263-018-1088-0

# Group Collaborative Representation for Image Set Classification

**Bo Liu[1,2] · Liping Jing[1] · Jia Li[1] · Jian Yu[1] · Alex Gittens[3] · Michael W. Mahoney[4,5]**

Received: 22 April 2016 / Accepted: 11 April 2018


**Abstract**
With significant advances in imaging technology, multiple images of a person or an object are becoming readily available in a number of real-life scenarios. In contrast to single images, image sets can capture a broad range of variations in the appearance of a single face or object. Recognition from these multiple images (i.e., image set classification) has gained significant attention in the area of computer vision. Unlike many existing approaches, which assume that only the images in the same set affect each other, this work develops a group collaborative representation (GCR) model which makes no such assumption, and which can effectively determine the hidden structure among image sets. Specifically, GCR takes advantage of the relationship between image sets to capture the inter- and intra-set variations, and it determines the characteristic subspaces of all the gallery sets. In these subspaces, individual gallery images and each probe set can be effectively represented via a self-representation learning scheme, which leads to increased discriminative ability and enhances robustness and efficiency of the prediction process. By conducting extensive experiments and comparing with state-of-the-art, we demonstrated the superiority of the proposed method on set-based face recognition and object categorization tasks.

**Keywords** Image set classification · Group collaborative representation · Point-to-sets representation · Set-to-sets representation

## 1 Introduction

With the development of image acquisition and transmission technologies, image set data is becoming increasingly available. One example of such a dataset is a collection of a person's facial images gathered from surveillance systems or from personal albums over a period of time; another example is a collection of multiple images of an object captured at different viewing angles by a network of cameras. For each person or object, the collection of images may form one or several image sets. The problem of classifying such image sets is attracting increasing interest in the computer vision and machine learning communities (Kim et al. 2007; Harandi et al. 2011; Lu et al. 2013; Hayat et al. 2014, 2015; Lu et al. 2015). Unlike traditional classification techniques based on single images, image set classification techniques

Communicated by M. Hebert.

✉ Liping Jing
lpjing@bjtu.edu.cn

Bo Liu
boliu@hebau.edu.cn

Jia Li
jail@bjtu.edu.cn

Jian Yu
jianyu@bjtu.edu.cn

Alex Gittens
gittea@rpi.edu

Michael W. Mahoney
mmahoney@stat.berkeley.edu

[1] Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China

[2] College of Information Science and Technology, Agricultural University of Hebei, Baoding 071000, Hebei, China

[3] Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY 12180, USA

[4] International Computer Science Institute, University of California at Berkeley, Berkeley, CA 94702, USA

[5] Department of Statistics, University of California at Berkeley, Berkeley, CA 94702, USA

Published online: 23 May 2018

estimate the label of a probe (or testing) set given a number of gallery (or training) sets.

In comparison to single images, image sets can incorporate a broad range of variations in the appearance of a single object, due to camera pose changes, non-rigid deformations, or simply different lighting conditions. This information is both potentially useful and a source of complex structure; the challenges of image set classification lie in modeling the structure inherent in image sets, and meaningfully measuring the similarity and differences between multiple sets.

Some approaches to modeling image sets include probabilistically modeling each image set with a Gaussian distribution (Arandjelovic et al. 2005; Shakhnarovich et al. 2002), and also representing the sets variously with linear subspaces (Kim et al. 2007; Harandi et al. 2011), exemplars or their affine or convex hulls (Cevikalp and Triggs 2010; Hu et al. 2012; Yang et al. 2013; Zhu et al. 2014), or their covariance matrices (Wang et al. 2012). Such methods represent each image set with a single model, and can have difficulty adequately representing large intra-set variations (arising from, e.g., images taken of the same face at different times, under different lighting conditions, or when the person has different expressions). In such cases where the set structure is complex, single-model methods are inadequate. Recently, researchers have proposed alternative multi-model approaches (Wang et al. 2008, 2015; Wang and Chen 2009; Chen et al. 2013) where each set is divided into several clusters and each cluster is modeled with a local linear subspace or a Gaussian distribution.

Although set-based representation methods have achieved promising performance, most retain the inherent limitation that they exclusively model intra-set structure and do not consider the relationships between images from different sets. However, one fact in face recognition is that the face images from different sets still have similarity (sets belonging to one person) and difference (sets belonging to different persons). Thus, it is useful to model the local structure among images within the same set and the relationships between different sets. Another practical consequence of this limitation is that these methods suffer when the image sets contain only a few images. In this case, because of the paucity of training data in the under-represented sets, it is difficult to capture their intra-set structure well enough to effectively model these sets; this leads to a decrease in the classification performance.

In addition to the choice of representation of the image sets, the choice of set similarity or distance measure is an important factor in the success of image set classification. Most previous methods have used the nearest neighbor scheme (Hu et al. 2012; Chen et al. 2013), which ignores the relationships among gallery sets (training data). Zhang et al. (2011) proposed a collaborative representation classification mechanism (CRC) for the single image classification problem that represents each image with the training images from all classes, thereby providing an alternative way to consider the relationships among the classes during the classification task. CRC has been empirically shown to be successful for the facial recognition task. The recent image set collaborative representation and classification (ISCRC) model of Zhu et al. (2014) extends CRC to the image set classification problem. However, ISCRC characterizes each image set with a single exemplar, and therefore has the same limited ability as other single-model representations to capture the complexity in image sets.

In this paper, we propose a group collaborative representation approach (GCR) that aims to model both the variations in each image set and the essential relationships among image sets. GCR consists of two related representations, as depicted in Fig. 1: point-to-sets representations (PSsR) for the individual images in the gallery sets, and set-to-sets representations (SSsR) for each probe set. To form these representations, GCR first learns a multi-model representation for each gallery set by employing spectral clustering on the self-expressive coefficients of the images within each gallery set (Lu et al. 2012). Each gallery set is then characterized by the collection of mean vectors of the images within each of its clusters. The PSsR representations express each training image as a linear combination of the collection of all local models across all the gallery sets; a convex combination of the group $\ell_1$ and $\ell_2$ regularizations (i.e., the group lasso and ridge penalties) is employed to encourage representations that involve all the gallery sets while simultaneously respecting the natural group structure inherent in the gallery sets. Classifiers fit using PSsR representations gain from the strengths of this collaborative multi-model representation as well as the abundance of training data, since these representations are available for each image in the training sets. Complimentarily, SSsR represents an entire probe set (as opposed to each image within the probe set) in terms of all the gallery sets in a similarly regularized manner. By thus separating the representations used at the training and testing phases of classification, the GCR scheme reduces the computational burden involved at the testing phase.

In the example shown in Fig. 2, the PSsR representation enhances the separability of the gallery sets (from Fig. 2a–c). Similarly, the proposed SSsR representation (shown in Fig. 2d) more unambiguously associates the probe set in this example with the appropriate gallery sets than does the original representation (shown in Fig. 2b). This example illustrates the discriminative power of the GCR representations, which can be taken advantage of using most traditional classification methods by training on the PSsR representations and using the SSsR representations at test time.

The key to the performance of GCR lies in three areas. First, because it uses the local models from all the gallery sets as the dictionary with respect to which each training image is represented, information on both intra- and inter-set

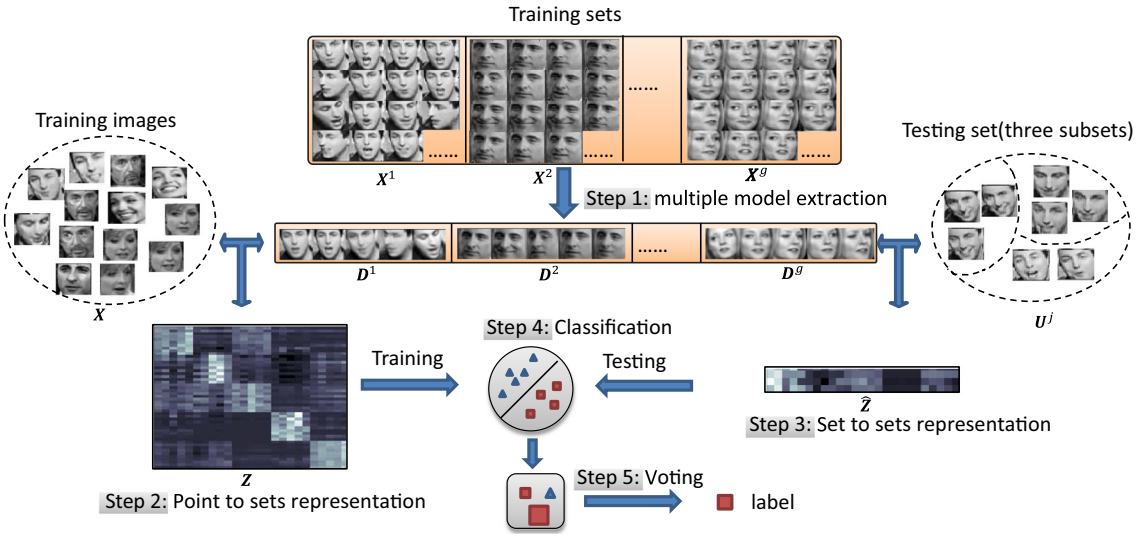


**Fig. 1** The framework of the proposed GCR method. In GCR, the image set classification can be implemented with five steps. In Step 1, the local structure is identified from each gallery set to form the dictionary $D$. Step 2 represents each gallery image with the proposed PSsR model. Each probe set is re-presented via the proposed SSsR model in Step 3. In Step 4, the classifier [such as ridge regression (RR) or kernelized version (KRR)] is trained on the new representation of gallery images. Step 5 predicts the label of each testing image set with the trained classifier and new representation of testing data

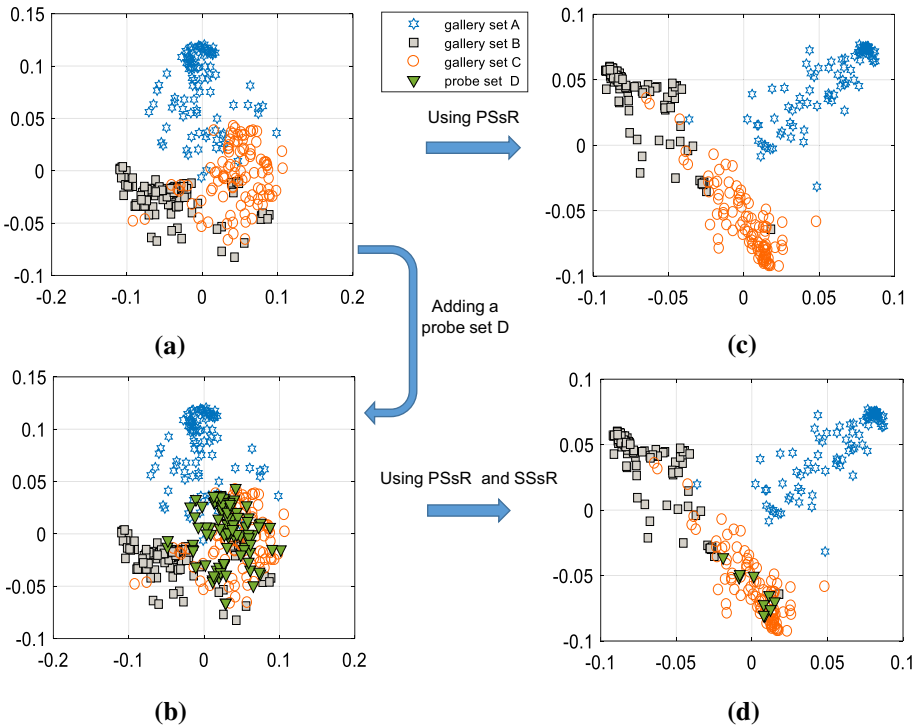


**Fig. 2** Representations of three subjects from the Honda face collection. Each gallery set contains 100 images selected from the corresponding subject, and the probe set D consists of 100 additional images of the subject in gallery set C. Panels **a** and **c** depict 2-dimensional PCA coordinates of the original image representation (pixels) and the PSsR representation of the three gallery sets respectively. Panel **b** superimposes the PCA representation of the probe set D onto (**a**), and panel **d** superimposes the SSsR representation of the probe set D (using a 10-cluster representation) onto (**c**)

variations are captured in the PSsR representations. Second, the group lasso regularization encourages sparsity of the PSsR and SSsR representations at the group level and is therefore helpful in identifying the most related gallery sets for each image or probe set, which increases the discriminative power of these representations. Third, the ridge penalty attempts to distribute the energy of the representations over all the local models, thereby increasing the stability of the representations. Theoretically, this combination of regularizations has the salutatory effect of guaranteeing that similar images have similar representations and two similar gallery sets contribute similarly to the representation of a given image.

GCR can also mitigate image set sparsity issues caused by the presence of gallery sets with few images. This is due to the fact that GCR (via PSsR), unlike existing single-model and multi-model methods, represents every gallery image rather than simply each gallery set. This, in combination with the way the PSsR representation encourages the sharing of information across gallery sets, assists in the modeling of gallery sets with fewer images and provides more information during the subsequent classification process. This is an important advantage to the GCR model, as in realistic applications we often do not have the opportunity to collect as many samples from each image set as we would like.

The remainder of this work is organized as follows. In Sect. 2 we review related works and we present the details of the group collaborative representation model in Sect. 3. In Sect. 3 we expound the optimization procedures for solving PSsR and SSsR model, and in Sect. 4 we outline the image set classification procedure using GCR representations. Section 5 presents comprehensive experiments conducted on six benchmark datasets: the results indicate that the GCR model is computationally efficient and consistently achieves better performance than eleven state-of-the-art techniques. Section 6 concludes with final remarks.

## 2 Related Works

Two issues must be addressed when performing image set classification: set representation, and the measurement of relationships between sets. Existing image set classification methods can be roughly divided into the categories of single-model and multi-model methods, based on the manner in which sets are represented. Single-model methods represent each set with a single model, while multi-model methods use multiple models.

Some single-model methods model image sets with parametric probability distributions (e.g., Gaussians) (Shakhnarovich et al. 2002; Arandjelovic et al. 2005) and measure dissimilarity between image sets with the Kullback–Leibler divergence. Such methods suffer when the gallery and probe sets do not exhibit any strong statistical relationship. In order to avoid parameter estimation, other single-model methods instead model each image set via a single linear subspace that is selected to capture the intra-set variance (Kim et al. 2007). Typically, this subspace is selected to be the dominant eigenspace of the image set covariance matrix and the distance between image sets is taken to be the principal angles between their subspaces. In a refinement of this model, Harandi et al. model image sets as points on a Grassmannian manifold that is chosen to capture both intra-class compactness and inter-class separability; various similarity measures can then be applied to the final Grassmannian representation to define the similarity of image sets (Harandi et al. 2011).

Although computationally attractive, linear subspace modeling does not perform well when the image set has only a few members or exhibits large and complex variations (Wang et al. 2012). By nature, they also capture only very weak information about the boundaries of image sets, which negatively impacts their discriminative power (Cevikalp and Triggs 2010). Consequently, researchers have also considered using representations based on the affine or convex hull of image sets; the distance between image sets is then measured as the distance between the appropriate hulls. Cevikalp and Triggs (2010) introduce affine and convex hull representations and find the distances by solving convex optimization problems. Hu et al. (2012) use the Cevikalp–Triggs affine hull representation, but take the distance between image sets to be the distance between two points in the sparse linear spans of the respective image sets; an optimization problem is solved to balance the sparsity with the closeness of the points. Yang et al. (2013) represent image sets as the intersection of their affine hull and an $\ell_p$ ball of specified radius, and take the distance between image sets to be the Euclidean distance between their representations. These methods provide a more expressive alternative to linear models, but incur significantly higher computational costs and rely on the assumption that image sets can be modelled as simple geometric structures.

More recently, Wang et al. (2012) proposed modeling each image set with its covariance matrix, and Lu et al. (2013), Uzair et al. (2014) extended this representation by additionally including the mean and the outer-product of the covariance matrix with the mean. To quantify set similarities in Wang et al. (2012), the covariance matrices are mapped from the manifold of positive-semidefinite matrices to a Euclidean space, and the Log-Euclidean distance is used as the distance metric. In Lu et al. (2013), a local kernel metric learning method is used to learn an appropriate similarity function, and in Uzair et al. (2014), a sparse kernel learning technique is used to learn a discriminative combination of a small number of several candidate kernels including the Log-Euclidean distance kernel. More generally, several metric learning methods have been proposed to find efficacious set-to-set and point-to-set metrics (Zhu et al. 2013; Huang et al. 2014) when sets are modeled as points on a Riemannian manifold.

The above models attempt to capture each image set using a single model, so can be expected to have trouble accurately capturing image sets with large internal variations. To mitigate this situation, multi-model approaches have been proposed that extract multiple local models from each image set via clustering, linear patch constructions, or joint sparse approximation (Wang et al. 2008; Wang and Chen 2009; Chen et al. 2013). Wang et al. (2015) recently extended this line of work by modeling image sets as Gaussian mixtures and using kernelized discriminant analysis to perform face recognition with image sets.

These multi-model approaches have demonstrated promising performance on image set classification, but ignore the relationships between image sets when representing image sets. In fact, using information from other image sets within the same class when learning the representation of a given image set may increase the discriminative power of the learned representation. As an example, the Template Deep Reconstruction Model of Hayat et al. (2015) learns a template for each class by fitting a deep neural network using all of the image sets from within that class, and classification is carried out by voting based on the reconstruction errors of all the templates when applied to a probe image set. The semi-supervised clustering framework of Mahmood et al. (2014) clusters all gallery and probe images simultaneously, thereby learning the distribution over the clusters for both the probe image set and the classes represented in the gallery; to classify, the probe image set is assigned to the class whose distribution over the clusters most resembles that of the probe set. Zhu et al. (2014) introduced a collaborative representation approach in which the probe set is represented as a point in the convex hull of the gallery images, but this method is limited by the fact that each image set is represented as a single point, and all gallery sets are treated equally in learning the representations.

Unlike previous works, in this work we represent each gallery image and each probe set using all the gallery data, and the influence of each gallery set on the representation of a given image or set varies intelligently depending on its importance to the probe. We exploit the structure within each gallery set and the relationships between the gallery sets to learn group collaborative representations (GCRs) for both images and sets. The former we call point-to-sets representations (PSsRs), and use to represent gallery images, and the latter we call set-to-set representations (SSsRs) and we use to represent probe image sets. Because of the group collaborative regularization, the PSsR and SSsR representations are more discriminative than existing representations. Additionally, since both representations are compact, the training and testing procedures are efficient. Complete details of these representations are given in the next section.

## 3 Group Collaborative Representation

Let the gallery data be $\boldsymbol{X} = \{\boldsymbol{X}^j\}_{1\leq j\leq g} \in \mathbb{R}^{d\times N}$. Here, $g$ is the number of image sets and there are $N$ total gallery images. The $j$th gallery set $\boldsymbol{X}^j = \{\boldsymbol{x}_i^j\}_{1\leq i\leq n^j} \in \mathbb{R}^{d\times n^j}$ contains $n^j$ images (hence $N = \sum_{j=1}^{g} n^j$); here, $\boldsymbol{x}_i^j \in \mathbb{R}^d$ indicates the $i$th image in the $j$th gallery set. Similarly, $\boldsymbol{U} = \{\boldsymbol{U}^j\}_{1\leq j\leq p} \in \mathbb{R}^{d\times M}$ denotes the probe data with $p$ sets, each probe set $\boldsymbol{U}^j = \{\boldsymbol{u}_i^j\}_{1\leq i\leq m^j}$ has $m^j$ images, and $\boldsymbol{u}_i^j \in \mathbb{R}^d$ indicates the $i$th image in the $j$th probe set. The total number of probe images is $M = \sum_{j=1}^{p} m^j$.

Our method aims to characterize each gallery image and each probe set using all the gallery data, in order to best use the intra- and inter-set variations to improve the discriminative ability of the representation. To capture the complex structure in gallery set $\boldsymbol{X}^j$, we extract from it multiple subspaces and represent it with their subspace means. We then represent each gallery image and probe set with respect to the dictionary formed by the collection of subspace means from all the gallery sets. A group collaborative regularization is enforced on the coding coefficients to ensure that the most relevant gallery sets are selected to encode the input data. Furthermore, the group collaborative regularization assists in ensuring that data from the same set have similar representations, which is a critical property in representations used for image classification.

### 3.1 Multiple Model Extraction

Because of the processes that generate image sets (e.g., variation in lighting or pose in the case of facial image sets), the image set usually has complicated structure. Fortunately, one common observation is that the set is approximated well by the union of several low-dimensional subspaces (Vidal 2011), which motivates us to segment our gallery sets into several clusters of low-dimensional subspaces, and then use these subspace means to represent the set's local models.

Several subspace clustering methods have been proposed in the literature; their applicability and accuracy vary depending on the domain of application, but they usually fall into one of four classes: algebraic, iterative, statistical, and spectral clustering-based (Vidal 2011). Among these, spectral clustering-based methods have demonstrated the ability to capture both global and local structure from the self-expressive coefficients (Vidal 2011). Lu et al. proposed a least squares regression method (LSR) to efficiently compute the self-expressive coefficients which are then used as inputs to a spectral clustering algorithm to obtain the cluster centers and the cluster membership of each instance (Lu et al. 2012). We follow this method and formulate the multiple models extraction from gallery set $\boldsymbol{X}^j \in \mathbb{R}^{d\times n^j}$ as the ridge regression problem

$$\begin{aligned} &\min_{\boldsymbol{Z}} \|\boldsymbol{X}^j - \boldsymbol{X}^j\boldsymbol{Z}\|_F^2 + \lambda\|\boldsymbol{Z}\|_F^2. \\ &\text{s.t. } \operatorname{diag}(\boldsymbol{Z}) = 0 \end{aligned} \tag{1}$$

Here, $\boldsymbol{Z} = [z_1, z_2, \ldots, z_{n^j}] \in \mathbb{R}^{n^j\times n^j}$ is the matrix of self-expressive coefficients. Each entry $z_{ba}$ indicates the extent to which the $b$th image $\boldsymbol{x}_b^j$ contributes to the representation of the $a$th image $\boldsymbol{x}_a^j$. $\operatorname{diag}(\boldsymbol{Z}) \in \mathbb{R}^{n^j}$ is a vector ($\in \mathbb{R}^{n^j}$) and its component is the corresponding diagonal entry of $\boldsymbol{Z}$. The constraint $\operatorname{diag}(\boldsymbol{Z}) = 0$ is adopted here to avoid the trivial solution of (1). This model encourages the coefficients of

a group of correlated images to be approximately equal (Lu et al. 2012), which is helpful for clustering. An affinity matrix among the images is then defined as $(|\boldsymbol{Z}| + |\boldsymbol{Z}|^T)/2$. Here $\lambda$ is set to 1 in experiments.

After applying spectral clustering to the affinity matrix, one obtains the cluster membership of all the images in $\boldsymbol{X}^j$, denoted by $\boldsymbol{C}^j \in \mathbb{R}^{n^j \times c}$. When the $i$th image is assigned to the $r$th cluster, $C^j_{ir} = 1$, otherwise $C^j_{ir} = 0$. The subspace corresponding to the $r$th cluster is represented by the mean vector of its members, denoted by $\boldsymbol{d}^j_r = \frac{\sum_{i=1}^{n^j} C^j_{ir} \boldsymbol{x}^j_i}{\sum_{i=1}^{n^j} C^j_{ir}} \in \mathbb{R}^d$. Note that the number of clusters ($c$) is a pre-defined parameter. For simplicity, we use the same $c$ for all sets, i.e., we extract the same number of local models from all sets. The effect of $c$ is empirically demonstrated and discussed in Sect. 5.

After extracting the local models from the gallery sets, we can form the dictionary $\boldsymbol{D} = \{\boldsymbol{D}^j\} = \{\boldsymbol{d}^j_r\} \in \mathbb{R}^{d \times cg}$, where $1 \leq j \leq g$ indicates the gallery index and $1 \leq r \leq c$ indicates the index of the local model extracted from the corresponding gallery set. The dictionary $\boldsymbol{D}$ is divided into $g$ subdictionaries $\boldsymbol{D}^j$ corresponding to the $j$th gallery set. Replacing the original images with the means of subspaces is good way to extract the multi-model structure from each gallery set. This is reasonable because the subspace means can simultaneously remove the noisy or redundant information from each image set and capture its local multi-models of each set.

### 3.2 Point-to-Sets Representation (PSsR)

In this subsection, we introduce our group collaborative representation for individual images, the Points-to-Sets representation (PSsR).

In many previous works, regardless of whether sets use a single or multi-model representation, each image is represented by exactly one of the models (e.g., the nearest cluster center). This choice may be suboptimal, as an image may be better represented by a combination of multiple models. Further, as discussed in Zhang et al. (2011), data from different classes may share similarities.

Motivated by these considerations, rather than representing a given image only in terms of a single model from one gallery set, we represent it in terms of all the models from all the gallery sets. To avoid clutter, we use $\boldsymbol{x}$ to refer to gallery image $\boldsymbol{x}^j_i$ in the following sections. A first pass at a satisfactory representation of $\boldsymbol{x}$ is given by the minimizer of

$$\min_{\boldsymbol{z}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{z}\|_2^2. \tag{2}$$

Since the dictionary $\boldsymbol{D}$ contains the models from all the gallery sets, models from multiple classes can all potentially contribute to modeling $\boldsymbol{x}$. Popular alternative choices for the loss function include the logistic loss and hinge loss. However, it has been shown that the least squares loss function is universally Fisher consistent and shares the same population minimizer with the squared hinge loss function (Zou et al. 2008). The least square loss function is also more convenient computationally.

Recall that the dictionary $\boldsymbol{D}$ is naturally divided into subdictionaries corresponding to the different gallery sets. It is reasonable to expect that the support of the representation coefficients $\boldsymbol{z}$ of each image across $\boldsymbol{D}$ should reflect this grouping. In particular, the contributions of different galleries to a given image should vary in importance, so rather than treat the subdictionaries equally, we add a weighted group lasso term to (2) with weights chosen according to the expected relevance of each gallery set to this image:

$$\Omega_1(\boldsymbol{z}) = \lambda_1 \sum_{j=1}^{g} w_j \|\boldsymbol{z}_{G_j}\|_2. \tag{3}$$

Here, $\boldsymbol{w} = \{w_j\}_{j=1,\ldots,g} \in \mathbb{R}^g$ is the set of gallery weights and $G_j$ is the set of indices corresponding to the $j$th subdictionary in $\boldsymbol{D}$. More specifically, when

$$\boldsymbol{D} = \left\{\boldsymbol{d}^1_1, \boldsymbol{d}^1_2, \ldots, \boldsymbol{d}^1_c, \boldsymbol{d}^2_1, \boldsymbol{d}^2_2, \ldots, \boldsymbol{d}^2_c, \ldots, \boldsymbol{d}^g_1, \boldsymbol{d}^g_2, \ldots, \boldsymbol{d}^g_c\right\}, \tag{4}$$

the set of indices corresponding to the $j$th gallery set is

$$G_j = \{(j-1)c + 1, \ldots, (j-1)c + c\}. \tag{5}$$

This weighted group lasso term imposes sparsity at the gallery level, so as $\lambda$ increases, the number of gallery sets with non-zero contributions to the representation of the image decreases (Yuan and Lin 2006).

To choose the weights, we follow the local consistency assumption (Cai et al. 2011), and take $w_j$ to be the normalization of the average Euclidean distance between the image and the elements of the $j$th subdictionary:

$$w_j = \frac{\mathrm{Dist}_{\mathrm{avg}}(\boldsymbol{x}, \boldsymbol{D}^j)}{\sum_{i=1}^{g} \mathrm{Dist}_{\mathrm{avg}}(\boldsymbol{x}, \boldsymbol{D}^i)}, \tag{6}$$

where

$$\mathrm{Dist}_{\mathrm{avg}}(\boldsymbol{x}, \boldsymbol{D}^j) = \frac{1}{c}\sum_{r=1}^{c} \|\boldsymbol{x} - \boldsymbol{d}^j_r\|_2.$$

This choice of weights ensures that the group coefficients $\boldsymbol{z}_{G_j}$ approach zero if the image is far from the $j$th gallery set. This is a desirable property, as it ensures that images are only represented in terms of galleries which have some

relation to the image, thereby increasing the stability of the image representations.

On the other hand, solely using this group lasso term would violate our desire that the collaborative representation use all of the local models. To remedy this situation, we use the $\ell_2$-norm regularizer

$$\Omega_2(z) = \lambda_2 \|z\|_2^2. \tag{7}$$

This regularizer essentially imposes a Gaussian prior on the entries of the $z$, which penalizes sparse vectors and thereby helps spread the support of the coefficients out among all the models. Simultaneously, it encourages similar local models to contribute similar amounts to the representation of the image $\boldsymbol{x}$ (Zou and Hastie 2005).

Our final PSsR representation of $\boldsymbol{x}$ is the coefficient vector $z$ that minimizes the combination of (2), (3), and (7):

$$\min_z \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{D}z\|_2^2 + \lambda_1 \sum_{j=1}^{g} w_j \|z_{G_j}\|_2 + \lambda_2 \|z\|_2^2, \tag{8}$$

The parameters $\lambda_1$ and $\lambda_2$ determine the tradeoff between the fidelity and regularization terms.

The proposed model (8) has two theoretical properties. First, two images have similar representation vectors $z_i$ and $z_j$ if they have similar ridge regression coefficients against dictionary D; this is quantified in Theorem 1.

**Theorem 1** *Given two images $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, let*

$$\boldsymbol{\beta}_{RR}(\boldsymbol{x}_i; \lambda_2) = (\boldsymbol{D}^T\boldsymbol{D} + \lambda_2 \boldsymbol{I})^{-1}\boldsymbol{D}^T\boldsymbol{x}_i$$

*denote the ridge regression coefficients of $\boldsymbol{x}_i$ against $\boldsymbol{D}$ with regularization parameter $\lambda_2$. The PSsR representations $z_1$ and $z_2$ of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ satisfy*

$$\|z_1 - z_2\|_2 \leq \|\boldsymbol{\beta}_{RR}(\boldsymbol{x}_1; \lambda_2) - \boldsymbol{\beta}_{RR}(\boldsymbol{x}_2; \lambda_2)\|_2 + \frac{2\lambda_1}{\lambda_2}.$$

Second, if two image sets have similar ridge regression coefficients against dictionary $\boldsymbol{D}$, they will make similar contributions to the representation of any image; this is quantified in Theorem 2. These two stability properties make the PSsR a discriminative representation and contribute towards the improvement of the performance of classifiers which use PSsRs as inputs.

**Theorem 2** *Let $z$ denote the PSsR of an image $\boldsymbol{x}$ and let $\beta_{RR}(\boldsymbol{x}; \lambda_2)$ be as defined in Theorem 1. Each pair $(z_{G_j}, z_{G_k})$, consisting of the coefficients of the $j$th and $k$th gallery sets, satisfies*

$$\|z_{G_j} - z_{G_k}\|_2 \leq \|\boldsymbol{\beta}_{RR}(\boldsymbol{x}; \lambda_2)_{G_j} - \boldsymbol{\beta}_{RR}(\boldsymbol{x}; \lambda_2)_{G_k}\|_2 + \frac{\lambda_1}{\lambda_2}(w_j + w_k).$$

The dependence of the bounds in Theorems 1 and 2 on $\lambda_1/\lambda_2$ suggests that as this ratio decreases, both types of stability (similar images having similar representations, and similar galleries contributing similarly to the PSsR of an image) increase. Thus we expect that the optimal choice of $\lambda_1$ is smaller than the optimal choice of $\lambda_2$; this is empirically verified in "Appendix 3".

Proofs of the theorems are given in "Appendix 1".

### 3.3 Set-to-Sets Representation (SSsR)

From a practitioner's perspective, it is important that classification algorithms be both highly accurate and quickly applicable to probe sets. The PSsR representation scheme can certainly be used to build classifiers for the individual images in probe sets, however applying classifiers to each image in the probe sets would be time-consuming. Also, as our primary goal is to obtain a prediction at the set-level, the final prediction for the entire probe set requires an additional step such as voting; voting and similar aggregation strategies may be sensitive to noise and outliers in the individual images.

To avoid the difficulties just mentioned, we propose building classifiers that use a set-to-sets representation (SSsR) for the $i$th probe set ($\boldsymbol{U}^i \in \mathbb{R}^{d\times m_i}$ with $m_i$ images). To calculate the SSsR, we first group the images of the probe set into $c$ clusters using subspace clustering, so that each cluster corresponds to one subspace (see Sect. 3.1). Let $\boldsymbol{U}_r^i \in \mathbb{R}^{d\times m_r^i}$ denote the $m_r^i$ images belonging to the $r$th subspace, then $\boldsymbol{U}^i = [\boldsymbol{U}_1^i, \ldots, \boldsymbol{U}_c^i]$ and $\sum_{r=1}^{c} m_r^i = m^i$. Similarly to the PSsR, the SSsR of each subspace $\boldsymbol{U}_r^i$ expresses this subspace in terms of the entire dictionary $\boldsymbol{D}$ of gallery images by solving the optimization problem

$$\min_{z,\boldsymbol{y}} \|\boldsymbol{U}_r^i\boldsymbol{y} - \boldsymbol{D}z\|_2^2 + \lambda_1 \sum_{j=1}^{g} w_j \|z_{G_j}\|_2 + \lambda_2\|z\|_2^2 + \lambda_3\|\boldsymbol{y}\|_2^2$$
$$\text{s.t.} \quad \sum_{k=1}^{m_r^i} y_k = 1. \tag{9}$$

Here the dictionary $\boldsymbol{D}$ and group indices $G_j$ are defined in (4) and (5) respectively. The coefficient $w_j$ for a fixed cluster $\boldsymbol{U}$ is obtained by averaging the $w_j$ defined in (6) over the images in $\boldsymbol{U}$. Each subspace $\boldsymbol{U}$ is thus represented by $z$, and the SSsR of a certain point $\boldsymbol{U}\boldsymbol{y}$ in its affine hull. The ridge regression term $\|\mathbf{y}\|_2^2$ ensures that the chosen points balance between minimizing the SSsR representation objective and using all of the images in $\boldsymbol{U}_i$. Once the representations $z_r$ for the individual subspaces $\boldsymbol{U}_r^i$ have been learned, the SSsR representation of the probe set is $\boldsymbol{Z} = [z_1, \ldots, z_c] \in \mathbb{R}^{cg\times c}$. Because SSsR is a more stable representation of a

probe set, voting strategies become less sensitive to noise and outliers in SSsRs than in the PSsRs of the individual images (see Table 9 in Sect. 5.5). Appropriate values of the trade-off parameters for the four terms can be tuned via cross-validation.

### 3.4 Discussion

Our proposed group collaborative representation model first extracts subspaces from the gallery sets, then represents every gallery image and every probe set with the aid of these extracted multiple models (subspaces), using respectively the PSsR and SSsR representations. Thus, the GCR model can be understood as both a multi-model representation and a collaborative representation model.

Most existing multi-model methods aim to capture the intra-set structure (Wang et al. 2008; Wang and Chen 2009; Chen et al. 2013; Wang et al. 2015), i.e., they model each set with the information that the current set contains. Such models ignore the relationships between sets, even sets within the same class. Also, since the statistical information used to characterize the sets are drawn solely from that set (e.g. means and covariance matrices), the learned statistics can have low confidences if there are insufficiently many images in the set. Our GCR model addresses these two deficiencies of multi-model representations by building representations formed with the aid of all gallery sets.

Zhu et al. (2014) also used the idea of collaborative representation (Zhang et al. 2011) to represent each image set with all the gallery sets. This allows their model to capture and exploit the relationships between all the training data. However, they adopted a single model representation for each set; the resulting information loss degrades the performance of classifiers built using their representations.

The proposed GCR model uses a multi-model collaborative representation that captures both intra- and inter-set relationships. The group lasso and ridge penalty are used in fitting both PSsR and SSsR to promote democratic representations. These regularizers balance the need to use all of the training data when generating the representation with the need to use the most relevant training data. Thus the GCR representation captures more useful discriminative information than prior representations.

### 3.5 Optimization Procedure

The optimization problems (8) for PSsR and (9) for SSsR are convex and can be solved by various methods. For simplicity in dealing with the group lasso penalty, we employ the alternating direction multiplier method (ADMM) (Boyd et al. 2011) to find the optimal solution. More detail is given in "Appendix 2".

## 4 Image Set Classification Using the GCR Representations

Either of the two GCR representations could be used to train and apply image set classifiers, but they possess different relative advantages. In particular, the PSsR provides more detailed and abundant information about image sets than the SSsR, however the SSsR can be computed much more quickly and results in a more concise representation of image sets. These observations suggest training classifiers using PSsR, to maximize the amount of information available during the training process, and applying them to probe sets using SSsR, to reduce the application cost. We follow this prescription, with one exception noted below.

Using PSsR, each image in each gallery set is represented as a vector in $\mathbb{R}^{cg}$, therefore either existing set-based classification methods or traditional classification methods (supplemented by, e.g., voting) can be used to build image set classifiers. In prior works, the nearest neighbor approach was usually adopted to estimate the labels of a probe set. However, local classification methods like nearest neighbors ignore the global information implicit in the gallery sets. Accordingly, in the remainder of this work, we use ridge regression (RR) and its kernelized version (KRR) (Saunders et al. 1998) to build image set classifiers.

Let $\boldsymbol{Z} \in \mathbb{R}^{cg \times N}$ contain the PSsRs of $g$ gallery image sets comprising a total of $N$ images. Given that there are $L$ classes, let $\boldsymbol{F} \in \mathbb{R}^{N \times L}$ be the class indicator matrix of the images, so $F_{ij} = 1$ if the $i$th image belongs to the $j$th class and otherwise $F_{ij} = 0$. The RR and KRR models learn a classifier by solving the optimization problem

$$\min_{\boldsymbol{H}} \|\phi(\boldsymbol{Z})^T \boldsymbol{H} - \boldsymbol{F}\|_F^2 + \beta \|\boldsymbol{H}\|_F^2, \tag{10}$$

where the feature map $\phi$ maps from $\mathbb{R}^{cg}$ into $\mathbb{R}^p$ for some integer $p$. In the case of RR, $\phi(\boldsymbol{Z}) = \boldsymbol{Z}$, and the minimizer of this problem is

$$\boldsymbol{H} = (\boldsymbol{Z}\boldsymbol{Z}^T + \beta \boldsymbol{I})^{-1} \boldsymbol{Z}\boldsymbol{F}. \tag{11}$$

Defining the kernel matrix $\boldsymbol{K} = \phi(\boldsymbol{Z})^T \phi(\boldsymbol{Z})$, the solution for KRR is

$$\boldsymbol{H} = \phi(\boldsymbol{Z})(\boldsymbol{K} + \beta \boldsymbol{I})^{-1} \boldsymbol{F}. \tag{12}$$

In our experiments with KRR, we consider both the mean kernel (Uzair et al. 2014) and Riemannian kernel (Wang et al. 2012). For the new representation output by GCR, we calculate the mean and covariance matrix of the PSsRs of the subset containing the $i$th image for two kernels. To apply the trained classifiers to a probe set, we compute the SSsR $\hat{\boldsymbol{Z}} \in \mathbb{R}^{cg \times c}$ of the probe set and predict the class indicator

matrix $\hat{\boldsymbol{F}}$ of the probe set as

$$\hat{\boldsymbol{F}} = \hat{\boldsymbol{Z}}^T \boldsymbol{H} = \hat{\boldsymbol{Z}}(\boldsymbol{Z}\boldsymbol{Z}^T + \beta \boldsymbol{I})^{-1}\boldsymbol{Z}\boldsymbol{F} \tag{13}$$

for RR, and

$$\hat{\boldsymbol{F}} = \phi(\hat{\boldsymbol{Z}})^T \boldsymbol{H} = \boldsymbol{K}'(\boldsymbol{K} + \beta \boldsymbol{I})^{-1}\boldsymbol{F} \tag{14}$$

for KRR. Here $\boldsymbol{K}' = \phi(\hat{\boldsymbol{Z}})^T\phi(\boldsymbol{Z})$ measures the similarity between the probe set and the gallery sets. For the mean kernel,

$$K'_{ij} = \exp\left(\frac{-\|\hat{\boldsymbol{z}}_i - \boldsymbol{\mu}_j\|_2}{\sigma^2}\right) \tag{15}$$

where $\boldsymbol{\mu}_j$ is the mean of the PSsRs of the gallery images which belong to the same subset as the $j$th image.

KRR using the Riemannian kernel is the exception mentioned earlier to our prescription of using the PSsR during training and SSsR at test time. This is because for the Riemannian kernel

$$K'_{ij} = \mathrm{tr}\left(\log(\hat{\boldsymbol{\Sigma}}_i)\log(\boldsymbol{\Sigma}_j)\right),$$

where $\hat{\boldsymbol{\Sigma}}_i$ and $\boldsymbol{\Sigma}_j$ are the covariance matrices of the subset containing the $i$th probe vector and the subset containing the $j$th gallery image. SSsR provides only one vector for each of the $c$ subsets learned during the SSsR procedure, but several are required to compute a covariance matrix. Thus when using the Riemannian kernel for classification, we use the PSsR during both training and testing.

The predicted class matrix $\hat{\boldsymbol{F}} \in \mathbb{R}^{c\times L}$ provided by RR or KRR provides a soft prediction of the class for each of the $c$ subsets in the probe set. To merge these predictions to obtain a single class for the probe set, we adopt the weighted average voting scheme of Yang et al. (2013):

$$j = \arg\max_j \frac{1}{\delta_j}\left(\frac{1}{c}\sum_{r=1}^{c}\hat{\boldsymbol{F}}_{rj}\right). \tag{16}$$

The weight $\delta_j$ serves as a confidence measure for the accuracy of the prediction of the $j$th class; it is defined as the sum of the singular values of the PSsRs of the gallery images in the $j$th class, $\delta_j = \sum_i \sigma_i(\boldsymbol{Z}^{(j)})$. One expects that if $\delta_j$ is large, there is a large amount of variation in the images drawn from the $j$th class, so the confidence of the classifier for the $j$th class will be lower. Thus, this voting scheme weighs the predictions of the $j$th class inversely proportional to $\delta_j$.

## 5 Experiments

To demonstrate the performance of our proposed GCR model, a series of experiments are conducted on six real-world image datasets for two typical computer vision tasks: face recognition and object categorization.

### 5.1 Datasets

In experiments, six 2D image datasets including four face datasets and two object image datasets are used to evaluate the proposed model. In face datasets, the facial images extracted from each video clip form one image set. In object datasets, the images of an object constitute one image set.

**Honda/UCSD** (Lee et al. 2003) consists of 59 video sequences featuring 20 different subjects. Face images were obtained using the Viola–Jones face detector (Viola and Jones 2004) and resized to 20 × 20 pixels. Following Hu et al. (2012), Zhu et al. (2014), we processed the images using histogram equalization. One set per subject was randomly selected for training the classifiers (20 sets in total); the remaining 39 sets were used during testing.

**Mobo** (Gross and Shi 2001) is a human pose identification dataset containing 96 video sequences of 24 subjects. Face images were extracted and resized to 40 × 40 pixels. Following Zhu et al. (2014), Yang et al. (2013)the images are represented using LBP features. One set is randomly selected from each subject to use in training, and the remainder are used in testing.

**YouTube Celebrities (YTC)** (Kim et al. 2008) is a challenging video dataset containing 1910 video clips of 47 celebrities. The images of each subject were collected under varying lighting and with diverse facial expressions and poses. The Viola–Jones face detector was used to extract face images which were subsequently resized to 32 × 32 pixels. We used LBP histogram features to represent the images, as we found this choice enhances the performance of most of the compared methods. Each clip is considered as one image set. Following Zhu et al. (2014), Yang et al. (2013), Hu et al. (2012), for each subject, 3 sets are randomly selected for training and 6 sets for testing.

**YouTube Faces (YTF)** (Wolf et al. 2011) contains 3425 videos of 1595 subjects. Each video is taken as an image set. Following Hayat et al. (2015), only the 226 subjects with four or more videos are used. LBP features provided by the author are used to represent the images. We randomly selected 3 sets from each subject for training and use the rest for testing.

**Table 1** Summary of the datasets

| Datasets | Honda | Mobo | YTC | YTF | ETH-80 | RGB-D |
|---|---|---|---|---|---|---|
| Classes (c) | 20 | 24 | 47 | 226 | 8 | 51 |
| Sets/c | 1–5 | 4 | 9 | 4–6 | 10 | 3–14 |
| Images/set | 13–618 | 202–897 | 7–350 | 48–2175 | 41 | 99–172 |
| Gallery sets/c | 1 | 1 | 3 | 3 | 5 | 3 |
| Probe sets/c | 0–4 | 3 | 6 | 1–3 | 5 | 0–11 |

**ETH-80** contains images representing 8 object categories. Each category is divided into 10 sub-categories, each of which contains 41 multi-view images resized to $32 \times 32$ pixels. The images are represented using LBP features. In our experiments, each subcategory is taken as a image set. We randomly selected 5 sets of each of the subcategories for training and used the remaining 5 sets for testing.

**RGB-D** (Lai et al. 2011) contains RGB and depth video sequences corresponding to 51 common household objects, taken from multiple viewing angles. Each multi-frame video sequence is taken as an image set. Each image is resized to $32 \times 32$ pixels and its intensity is used as the input feature representation. We randomly selected 3 sets from each object for training and used the remaining sets for testing.

The data sets are summarized in Table 1. The number of classes per dataset varies between 8 and 226. Usually classification is harder for datasets with more classes, so in particular, facial recognition on the YTF dataset is a challenging task. Also, the number of images in each set varies by a lot both between datasets and within the classes of each dataset; this further affects the performance of most existing image set classification methods, as fewer images provide less information to be used in modeling the set. For each dataset, the training and testing subsets are randomly generated 10 times and the average results are reported. When the number of probe sets is zero for one class, it means that there is no testing data in the corresponding class.

## 5.2 Methodology

We compare the performance of our proposed method to eleven existing methods. Eight of the baseline methods are single-model methods; the remaining three are multi-model methods.

Of the single-model methods, five take exemplar-based approaches in which each image set is represented by an exemplar:

- AHISD (Cevikalp and Triggs 2010),
- CHISD (Cevikalp and Triggs 2010),
- SANP (Hu et al. 2012),
- RNP (Yang et al. 2013), and
- ISCRC (Zhu et al. 2014);

and three take structure-based approaches in which each set is represented as an affine/convex hull, or as a subset of a Riemannian manifold:

- DCC (Kim et al. 2007),
- CDL (Wang et al. 2012), and
- SSMDL (Zhu et al. 2013).

The multi-model methods considered are

- MMD (Wang et al. 2008),
- MDA (Wang and Chen 2009), and
- DARG (Wang et al. 2015).

The codes for all these methods were available on the authors' websites, with the exception of DARG. Due to there being no publicly available code, we implemented DARG in MATLAB.

These single-model methods have several hyperparameters: for DCC, the dimension of each subspace is set to 10; for CDL, Partial Least Squares is used as the classifier; for SSMDL, the number of positive pairs per set is selected from $\{1, 3, 5\}$ and the number of negative pairs per set is selected from $\{3, 5, 10\}$; for AHISD and CHISD, the percentage of energy preserved by PCA is set to 90 and the loss penalty parameter $C$ is set to 100. To determine the parameters used in SANP, RNP and ISCRC to balance between the loss function and the regularization term, we used values from the set $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$ and recorded the best results. For ISCRC, the number of atoms per set is selected from $\{5, 10, 20\}$.

For the multi-model methods MMD, MDA and DARG, the number of local models is selected from $\{1, 3, 5, 7\}$ and the best results are reported. The percentage of energy preserved by PCA and the distance ratio for MMD are set following Wang et al. (2008). For MDA, the number of between-class neighboring local models is set to 5. The fusing coefficients used in DARG for combining kernels derived from the Mahalanobis and Log-Euclidean distances are chosen following Wang et al. (2015).

We built RR and KRR classifiers using the proposed GCR model. These classifiers are denoted with GCR for classifiers built using RR on top of GCR, GCR(m) for classifiers built

**Table 2** Classification accuracy (average ± standard deviation) of GCR and baseline methods

| Methods | Honda | Mobo | YTC | YTF | RGB-D | ETH |
|---|---|---|---|---|---|---|
| RR | 93.08 ± 2.43 | 97.08 ± 1.02 | 69.22 ± 3.97 | 36.37 ± 1.12 | 62.95 ± 3.97 | 88.75 ± 4.75 |
| GCR | 98.72 ± 1.81 | **98.61 ± 0.65** | 77.80 ± 3.50 | **53.86 ± 2.92** | 72.86 ± 2.91 | 92.12 ± 3.99 |
| KRR(m) | 97.95 ± 2.02 | 97.64 ± 0.67 | 69.22 ± 3.97 | 49.39 ± 1.96 | 75.00 ± 3.41 | 91.75 ± 4.42 |
| GCR(m) | **99.74 ± 0.81** | 98.19 ± 0.93 | **79.30 ± 2.92** | 52.22 ± 2.21 | 70.37 ± 3.41 | 90.25 ± 3.74 |
| KRR(r) | **99.74 ± 0.81** | 92.36 ± 2.10 | 61.67 ± 3.57 | 37.78 ± 1.83 | 79.80 ± 2.83 | 92.75 ± 4.16 |
| GCR(r) | **99.74 ± 0.81** | 93.33 ± 2.51 | 63.64 ± 3.31 | 42.23 ± 2.58 | **83.06 ± 1.88** | **96.50 ± 3.07** |

using KRR with the mean kernel, and GCR(r) for classifiers built using KRR with the Riemannian kernel. The effect of parameters on GCR has been tested and the detail is given in "Appendix 3".

Classification performance is evaluated via accuracy. Given $p$ probe sets with ground truth label $l_i$ for the $i$th probe set and corresponding predicted label $f_i$, the accuracy is defined via

$$Accuracy = \frac{\sum_{i=1}^{p} \delta(l_i, f_i)}{p}. \tag{17}$$

Here $\delta$ is the Kronecker delta: $\delta(x, y) = 1$ if $x = y$, otherwise $\delta(x, y) = 0$.

### 5.3 Integrating Hand-Crafted Features with GCR

In this section, three experiments are conducted to demonstrate the performance of GCR with the aid of hand-crafted features.

#### 5.3.1 Comparing with Single Image-Based Classifiers (RR and KRR)

In this subsection, two single image-based classifiers, RR and KRR, are used as baselines to demonstrate the performance of the proposed GCR framework. The label of a probe set is predicted by a majority voting scheme. For each data set, all images are used to train the classifier.

The results are shown in Table 2, where GCR is based on RR, and GCR(m) and GCR(r) are based on KRR with the mean kernel and the Riemannian kernel respectively. The best results are in bold, and the second-best results are underlined. As expected, the proposed GCR framework outperforms the corresponding baseline. This result confirms that considering intra-set local structure and inter-set relationship is helpful to construct the new representation for the gallery images and probe sets and further enhance the set-based image classification accuracy.

**(a)**

**(b)**

**Fig. 3** Face examples from the Honda and Mobo datasets. **a** Honda, **b** Mobo

**(a)**

**(b)**

**Fig. 4** Face examples from the YTC and YTF datasets. **a** YTC, **b** YTF

#### 5.3.2 Set-Based Face Recognition

In this subsection, we compare the performance of our proposed method with that of eleven existing methods on the task of set-based face recognition. Set-based face datasets are commonly extracted from personal videos or surveillance recordings. Each set of images consists of consecutive frames featuring one person's face.

Figures 3 and 4 show examples of sets of face images corresponding to different subjects. It can be seen that, because

**Table 3** Classification accuracy (average ± standard deviation) of fourteen methods on the Honda dataset

| Methods | 50–50 | 100–100 | All |
|---|---|---|---|
| DCC (Kim et al. 2007) | 77.09 ± 3.34 | 83.08 ± 3.23 | 92.01 ± 3.21 |
| CDL (Wang et al. 2012) | 97.95 ± 1.62 | 99.49 ± 1.08 | 99.49 ± 1.08 |
| SSDML (Zhu et al. 2013) | 89.23 ± 4.49 | 88.46 ± 4.05 | 89.61 ± 3.89 |
| AHISD (Cevikalp and Triggs 2010) | 93.85 ± 3.66 | 93.97 ± 2.62 | 93.81 ± 2.97 |
| CHISD (Cevikalp and Triggs 2010) | 90.77 ± 3.46 | 92.92 ± 5.37 | 94.62 ± 2.54 |
| SANP (Hu et al. 2012) | 92.82 ± 3.97 | 93.85 ± 3.66 | 91.79 ± 3.37 |
| RNP (Yang et al. 2013) | 96.67 ± 3.02 | 96.51 ± 2.86 | 96.92 ± 2.64 |
| ISCRC (Zhu et al. 2014) | 97.95 ± 2.35 | 98.46 ± 1.79 | 98.97 ± 1.73 |
| MMD (Wang et al. 2008) | 87.94 ± 3.42 | 88.20 ± 3.86 | 90.25 ± 2.35 |
| MDA (Wang and Chen 2009) | 88.72 ± 4.04 | 90.26 ± 5.51 | 92.82 ± 4.80 |
| DARG (Wang et al. 2015) | 96.41 ± 1.93 | 98.72 ± 2.24 | 99.23 ± 1.05 |
| GCR | 98.72 ± 1.81 | **99.74 ± 0.81** | 98.72 ± 1.81 |
| GCR(m) | 98.97 ± 2.47 | 98.97 ± 1.32 | **99.74 ± 0.81** |
| GCR(r) | **99.49 ± 1.08** | **99.74 ± 0.81** | **99.74 ± 0.81** |

**Table 4** Classification accuracy (average ± standard deviation) of fourteen methods on the Mobo dataset

| Methods | 50–50 | 100–100 | All |
|---|---|---|---|
| DCC (Kim et al. 2007) | 88.30 ± 5.41 | 90.69 ± 3.01 | 91.25 ± 1.61 |
| CDL (Wang et al. 2012) | 82.36 ± 3.34 | 85.89 ± 2.94 | 88.86 ± 3.10 |
| SSDML (Zhu et al. 2013) | 95.95 ± 2.46 | 96.62 ± 2.40 | 97.03 ± 1.77 |
| AHISD (Cevikalp and Triggs 2010) | 96.35 ± 2.30 | 96.89 ± 1.28 | 97.70 ± 1.43 |
| CHISD (Cevikalp and Triggs 2010) | 95.68 ± 2.78 | 97.30 ± 1.68 | 96.76 ± 1.30 |
| SANP (Hu et al. 2012) | 91.38 ± 3.91 | 92.08 ± 3.82 | 97.92 ± 1.18 |
| RNP (Yang et al. 2013) | 96.81 ± 1.27 | 97.78 ± 1.68 | 98.06 ± 1.87 |
| ISCRC (Zhu et al. 2014) | 96.49 ± 1.77 | 97.81 ± 1.42 | 97.79 ± 1.24 |
| MMD (Wang et al. 2008) | 91.39 ± 3.91 | 92.08 ± 3.82 | 92.50 ± 3.53 |
| MDA (Wang and Chen 2009) | 93.06 ± 1.38 | 94.44 ± 3.14 | 96.66 ± 1.63 |
| DARG (Wang et al. 2015) | 96.94 ± 1.57 | 97.64 ± 1.93 | 97.62 ± 0.87 |
| GCR | **98.19 ± 1.32** | **98.47 ± 1.02** | **98.61 ± 0.65** |
| GCR(m) | 97.78 ± 1.49 | 97.92 ± 1.63 | 98.19 ± 0.93 |
| GCR(r) | 88.47 ± 2.70 | 93.06 ± 3.33 | 93.33 ± 2.51 |

the videos in the YTC and YTF datasets were recorded in unconstrained environments, each face set exhibits a large variance in lighting conditions, poses, and expressions. Accordingly, face recognition in the YTC and YTF datasets is more challenging than face recognition in the Hondo and Mobo datasets.

To evaluate the effect of set size on these methods, we measured the classification accuracy while varying the face set size for both the gallery and probe sets between 50 images, 100 images, and all available images. Sets which contain less than 50 frames are always used in their entirety.

The 10-fold cross-validated average and standard deviation of the classifier accuracies on the Honda and Mobo datasets are recorded in Tables 3 and 4 respectively. Although most of the methods achieve comparable performances on these two datasets, classifiers based on our proposed GCR model consistently obtain the best results. More specifically, Riemannian KRR using GCR representations as input features [i.e., GCR(r)] outperforms the other methods on the Honda dataset, and RR using GCR representations as input features (i.e., GCR) exhibits the best performance on the Mobo dataset.

The prediction results on the YTC and YTF datasets are listed in Tables 5 and 6 respectively. As expected, even simply using ridge regression, our proposed GCR representations significantly outperform the existing set-based image classification methods. This result confirms that the group collaborative representations more successfully capture the structure of the gallery and probe image sets than existing single-model and multi-model representation methods.

**Table 5** Classification accuracy (average ± standard deviation) of fourteen methods on the YTC dataset

| Methods | 50–50 | 100–100 | All |
|---|---|---|---|
| DCC (Kim et al. 2007) | 66.64 ± 4.41 | 68.37 ± 3.84 | 69.12 ± 3.81 |
| CDL (Wang et al. 2012) | 47.29 ± 3.57 | 54.43 ± 4.24 | 55.73 ± 4.36 |
| SSDML (Zhu et al. 2013) | 74.08 ± 3.87 | 74.98 ± 3.96 | 75.38 ± 3.34 |
| AHISD (Cevikalp and Triggs 2010) | 72.18 ± 3.16 | 72.98 ± 3.02 | 73.42 ± 2.78 |
| CHISD (Cevikalp and Triggs 2010) | 73.54 ± 2.93 | 73.44 ± 3.61 | 73.73 ± 3.90 |
| SANP (Hu et al. 2012) | 72.20 ± 2.91 | 73.31 ± 3.52 | 73.61 ± 3.36 |
| RNP (Yang et al. 2013) | 74.98 ± 3.96 | 75.38 ± 3.34 | 74.08 ± 3.87 |
| ISCRC (Zhu et al. 2014) | 75.33 ± 3.70 | 75.48 ± 4.10 | 76.33 ± 2.91 |
| MMD (Wang et al. 2008) | 71.06 ± 4.73 | 71.27 ± 3.55 | 71.13 ± 3.14 |
| MDA (Wang and Chen 2009) | 75.76 ± 3.50 | 75.38 ± 2.95 | 75.82 ± 3.95 |
| DARG (Wang et al. 2015) | 76.36 ± 3.43 | 76.56 ± 3.61 | 76.98 ± 3.05 |
| GCR | 77.67 ± 3.56 | 77.75 ± 3.82 | 77.80 ± 3.50 |
| GCR(m) | **78.80 ± 2.84** | **79.18 ± 3.10** | **79.30 ± 2.93** |
| GCR(r) | 55.28 ± 3.77 | 61.64 ± 3.29 | 63.64 ± 3.31 |

**Table 6** Classification accuracy (average ± standard deviation) of fourteen methods on the YTF dataset

| Methods | 50–50 | 100–100 | All |
|---|---|---|---|
| DCC (Kim et al. 2007) | 30.92 ± 1.49 | 30.64 ± 1.30 | 32.15 ± 2.06 |
| CDL (Wang et al. 2012) | 37.13 ± 2.03 | 38.31 ± 2.53 | 40.17 ± 1.55 |
| SSDML (Zhu et al. 2013) | 34.15 ± 1.57 | 35.32 ± 2.03 | 36.24 ± 2.05 |
| AHISD (Cevikalp and Triggs 2010) | 29.35 ± 0.83 | 31.57 ± 2.76 | 31.53 ± 1.65 |
| CHISD (Cevikalp and Triggs 2010) | 32.33 ± 1.95 | 32.59 ± 2.25 | 33.09 ± 1.84 |
| SANP (Hu et al. 2012) | 30.98 ± 2.49 | 31.48 ± 0.85 | 32.02 ± 1.32 |
| RNP (Yang et al. 2013) | 32.15 ± 2.73 | 34.41 ± 0.81 | 35.07 ± 1.39 |
| ISCRC (Zhu et al. 2014) | 38.77 ± 3.34 | 40.75 ± 0.62 | 41.97 ± 1.72 |
| MMD (Wang et al. 2008) | 32.33 ± 1.95 | 32.59 ± 2.25 | 34.56 ± 2.04 |
| MDA (Wang and Chen 2009) | 34.74 ± 2.40 | 34.26 ± 0.82 | 34.88 ± 0.97 |
| DARG (Wang et al. 2015) | 37.71 ± 2.00 | 39.63 ± 2.42 | 41.08 ± 1.97 |
| GCR | **53.01 ± 2.87** | **53.58 ± 2.95** | **53.86 ± 2.92** |
| GCR(m) | 51.67 ± 1.19 | 52.83 ± 2.98 | 52.22 ± 2.21 |
| GCR(r) | 37.20 ± 2.19 | 42.12 ± 1.99 | 42.23 ± 2.58 |

For the challenging dataset YTF, ISCRC is the best of the existing classification methods, which demonstrates that collaborative representation is beneficial in modeling these image sets. However, at training time ISCRC represents each gallery set as a whole in terms of the other gallery image sets; this may lose information retained in the PSsR representation, which represents every image in the gallery in terms of the other gallery image sets.

An interesting observation from this experiment is that GCR is relatively insensitive to the set size. For example, the multi-model method DARG and the single-model method CDL (the second best performing method on the Honda dataset) obtain better performance as the set size increases, while GCR(r) exhibits only a slight variance among the three settings (50, 100, All). This implies that GCR can work well when there are few images in the gallery or probe sets.

#### 5.3.3 Set-Based Object Categorization

In this subsection, we evaluate the methods on the multi-view object categorization task. In this task, each object is recorded in multiple images captured at different angles. We use two popular benchmark datasets, ETH and RGB-D, in our experiments. Figure 5 shows several examples of image sets drawn from ETH and RGB-D. The image sets in the ETH dataset contain 41 images, so we only conducted 'All' experiments, i.e., all images are used in the gallery and probe sets.

The classification results on ETH and RGB-D are listed in Table 7. It can be seen that GCR(r), CDL, and DARG are superior to the other methods, which is consistent with the results of Wang et al. (2012), Uzair et al. (2014). Among the methods, Riemannian KRR using GCR representations

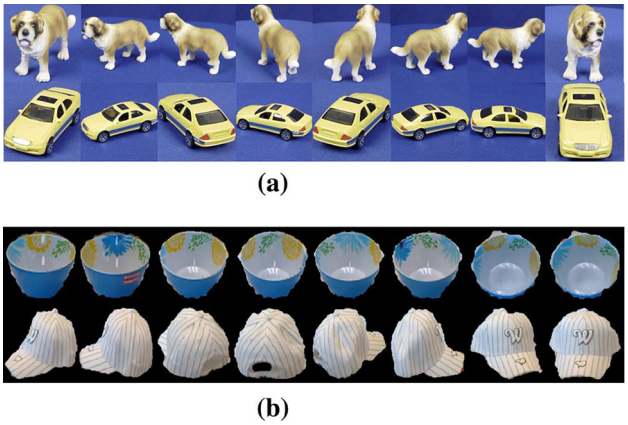

**Fig. 5** Object examples from the ETH and RGB-D datasets. **a** ETH, **b** RGB-D

is demonstrated to be the best choice for computing the similarity between two sets for the multi-view object categorization task. We note that both GCR(r) and CDL model each set with a covariance matrix. However, the set's covariance matrix used by CDL is obtained using only the information of the current set, and ignores the correlations between image sets. The GCR(r) covariance matrix, on the other hand, is built from a collaborative representation so implicitly takes into account the correlations between image sets. This difference contributes to the superior performance of GCR(r).

### 5.4 Integrating Deep Features with GCR

The proposed GCR framework can be constructed on different kinds of input features including hand-crafted features (e.g., LBP or intensity) and deep neural network-learned features (e.g., CNN network). In the previous experiments, we focus on the hand-crafted features. In this subsection, we conducted a series of experiments on YTC, YTF and RGB-D to compare the proposed GCR (taking deep features as input) with TDRM (Hayat et al. 2015) (an encoder–decoder neural network) and the existing single image-based deep convolution neural network classifiers.

The VGG16 CNN architecture (Simonyan and Zisserman 2014) is adopted here to learn the deep features. For YTF, the original CNN network is trained on a 2622 celebrities dataset (Parkhi et al. 2015) and fine-tuned by YTF. Since YTC and 2622 celebrities dataset share some common celebrities, we use ImageNet to train a VGG16 CNN network and fine-tune it with YTC and RGB-D respectively. Three end-to-end neural network classifiers based on VGG16 (for single image classification) are used as baselines. One outputs the label of each set via a majority voting scheme (denoted as VGG16). The other two exploit mean and max pooling schemes on the basic feature aggregation of all images in each set and obtain the corresponding label [denoted as VGG16(max) and VGG16(mean)].

The comparison results are listed in Table 8. From Table 8, we can get the following three observations. Firstly, the deep neural network-based features can always improve the performance of our proposed GCR model, i.e., VGG16-GCR is better than GCR on three datasets, where GCR is trained on the hand-crafted features (LBP for face datasets YTC and YTF, intensity for object dataset RGB-D). Secondly, good feature learning architecture is essential for deep neural network-based image set classification, which is why VGG16 outperforms TDRM, where the majority voting is adopted to predict the label of each probe set. Thirdly, it is not appro-

**Table 7** Classification accuracy (average ± standard deviation) of fourteen methods on the RGB-D and ETH datasets

| Methods | RGB-D | | | ETH |
|---|---|---|---|---|
| | 50-50 | 100-100 | All | All |
| DCC (Kim et al. 2007) | 65.58 ± 4.12 | 72.79 ± 1.66 | 71.16 ± 2.94 | 88.00 ± 4.83 |
| CDL (Wang et al. 2012) | 77.96 ± 3.67 | 78.71 ± 2.15 | 80.54 ± 3.24 | 94.75 ± 3.47 |
| SSDML (Zhu et al. 2013) | 62.72 ± 2.65 | 63.72 ± 3.22 | 66.53 ± 3.61 | 87.75 ± 5.45 |
| AHISD (Cevikalp and Triggs 2010) | 62.38 ± 3.12 | 62.93 ± 4.23 | 62.41 ± 1.70 | 79.50 ± 6.06 |
| CHISD (Cevikalp and Triggs 2010) | 63.05 ± 1.78 | 63.67 ± 1.84 | 64.84 ± 2.25 | 84.17 ± 4.78 |
| SANP (Hu et al. 2012) | 42.81 ± 3.05 | 43.29 ± 2.14 | 44.90 ± 2.15 | 83.50 ± 3.48 |
| RNP (Yang et al. 2013) | 61.56 ± 1.74 | 63.27 ± 3.86 | 65.90 ± 2.89 | 88.50 ± 4.89 |
| ISCRC (Zhu et al. 2014) | 60.08 ± 3.25 | 63.61 ± 3.13 | 65.17 ± 2.84 | 83.75 ± 6.32 |
| MMD (Wang et al. 2008) | 52.79 ± 2.57 | 54.29 ± 3.86 | 52.52 ± 1.00 | 84.85 ± 4.93 |
| MDA (Wang and Chen 2009) | 47.62 ± 2.84 | 60.68 ± 2.94 | 62.51 ± 3.04 | 86.75 ± 4.57 |
| DARG (Wang et al. 2015) | 76.93 ± 2.37 | 80.06 ± 2.75 | 81.08 ± 2.70 | 95.75 ± 3.53 |
| GCR | 70.15 ± 3.35 | 70.68 ± 2.65 | 72.86 ± 2.91 | 92.12 ± 3.99 |
| GCR(m) | 67.86 ± 3.64 | 69.83 ± 2.78 | 70.37 ± 2.99 | 90.25 ± 3.74 |
| GCR(r) | **78.37 ± 2.51** | **82.86 ± 1.77** | **83.06 ± 1.88** | **96.50 ± 3.07** |

**Table 8** Classification accuracy (average ± standard deviation) of GCR methods combined with CNN features

| Methods | YTC | YTF | RGB-D |
|---|---|---|---|
| VGG16(max) | 26.31 ± 1.96 | 26.09 ± 2.01 | 23.38 ± 4.08 |
| VGG16(mean) | 57.30 ± 2.05 | 70.64 ± 1.34 | 44.54 ± 2.93 |
| VGG16 | 81.50 ± 3.49 | 84.07 ± 3.09 | 79.46 ± 4.49 |
| TDRM | 77.56 ± 3.11 | 52.03 ± 2.67 | 69.83 ± 3.82 |
| GCR | 77.80 ± 3.50 | 53.86 ± 2.92 | 72.14 ± 3.95 |
| VGG16+GCR | 82.21 ± 3.02 | 86.28 ± 0.99 | 82.14 ± 3.95 |
| GCR(m) | 79.30 ± 2.93 | 52.22 ± 2.21 | 70.37 ± 2.99 |
| VGG16-GCR(m) | **82.88 ± 2.93** | **87.17 ± 1.17** | 82.23 ± 3.79 |
| GCR(r) | 63.64 ± 3.31 | 42.23 ± 2.58 | 83.06 ± 1.88 |
| VGG16-GCR(r) | 76.38 ± 2.83 | 58.87 ± 2.34 | **86.39 ± 2.17** |

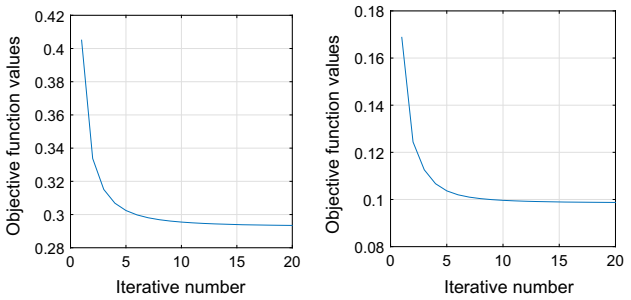


**Fig. 6** Objective function values of PSsR (**a**) and SSsR (**b**) versus iteration count for the Mobo dataset.

priate to use the max or mean pooling on the basic feature aggregation, because it may result in losing the structure information among each set.

## 5.5 Comparison of Running Time

To fit classifiers using the GCR model, one must obtain the PSsR representations for the gallery sets and the SSsR representations for the probe sets. The optimization problems (8) and (9) associated with the PSsR and SSsR models can be solved using ADMM as described in Sect. 3.5. The convergence of the ADMM algorithm for this use case, where the alternating minimization is with respect to two blocks of variables, has been well established (Boyd et al. 2011). In fact, because one of the two functions in the objectives of both the PSsR and SSsR is strictly convex, ADMM is guaranteed to exhibit a linear convergence rate (Deng and Yin 2012). Figure 6 shows the objective function values of both GCR representations as a function of the iterations, for the Mobo dataset using all images in each set.

Both algorithms converge in 20 iterations. In combination with the fact that the iterations can be completed by applying closed-form formulas involving nothing more computationally intensive than solving a linear system, these plots are evidence supporting the assertion that our proposed method finds reasonable representations in a small amount of time.

We advocated using SSsR to model the probe sets instead of PSsR for the sake of efficiency at test time. To demonstrate the effectiveness of this proposal, we conducted an experiment comparing using PSsR to model the Mobo probe sets versus using SSsR (using all images in each set). The prediction accuracy and the running time of the prediction phase are shown in Table 9. (Note that the training phases for RR + SSsR and RR + PSsR are identical.) As expected, SSsR shortens the prediction time both in terms of the time needed to find the probe set representations and the time needed to predict using these representations. It also increases the clas-

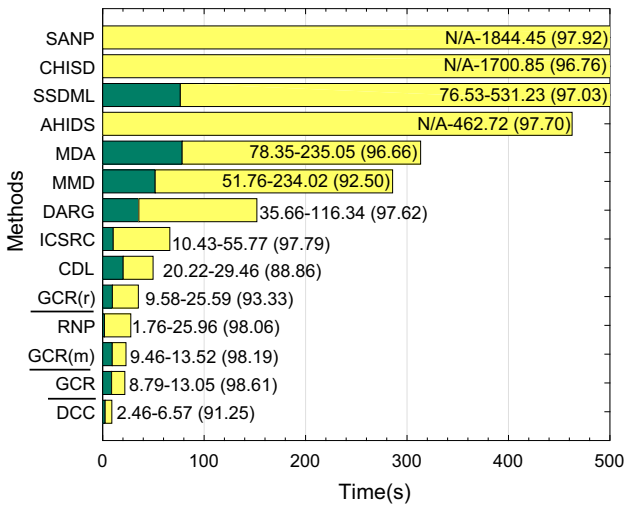


**Fig. 7** The running times (training in green, testing in yellow) of fourteen methods on the Mobo dataset, and the prediction accuracy on testing data (the value in parentheses for each method) (Color figure online)

**Table 9** Prediction performance and timing when using SSsR or PSsR to model probe sets with all images in the Mobo dataset

| Methods | Accuracy | Time (s) for generating SSsR or PSsR | Time (s) for prediction |
|---|---|---|---|
| RR+SSsR | 98.61 ± 0.65 | 13.05 | 1.43e−4 |
| RR+PSsR | 97.64 ± 1.14 | 24.93 | 0.12 |
| KRR(m)+SSsR | 98.19 ± 0.93 | 13.52 | 0.05 |
| KRR(m)+PSsR | 96.81 ± 1.16 | 25.84 | 0.13 |

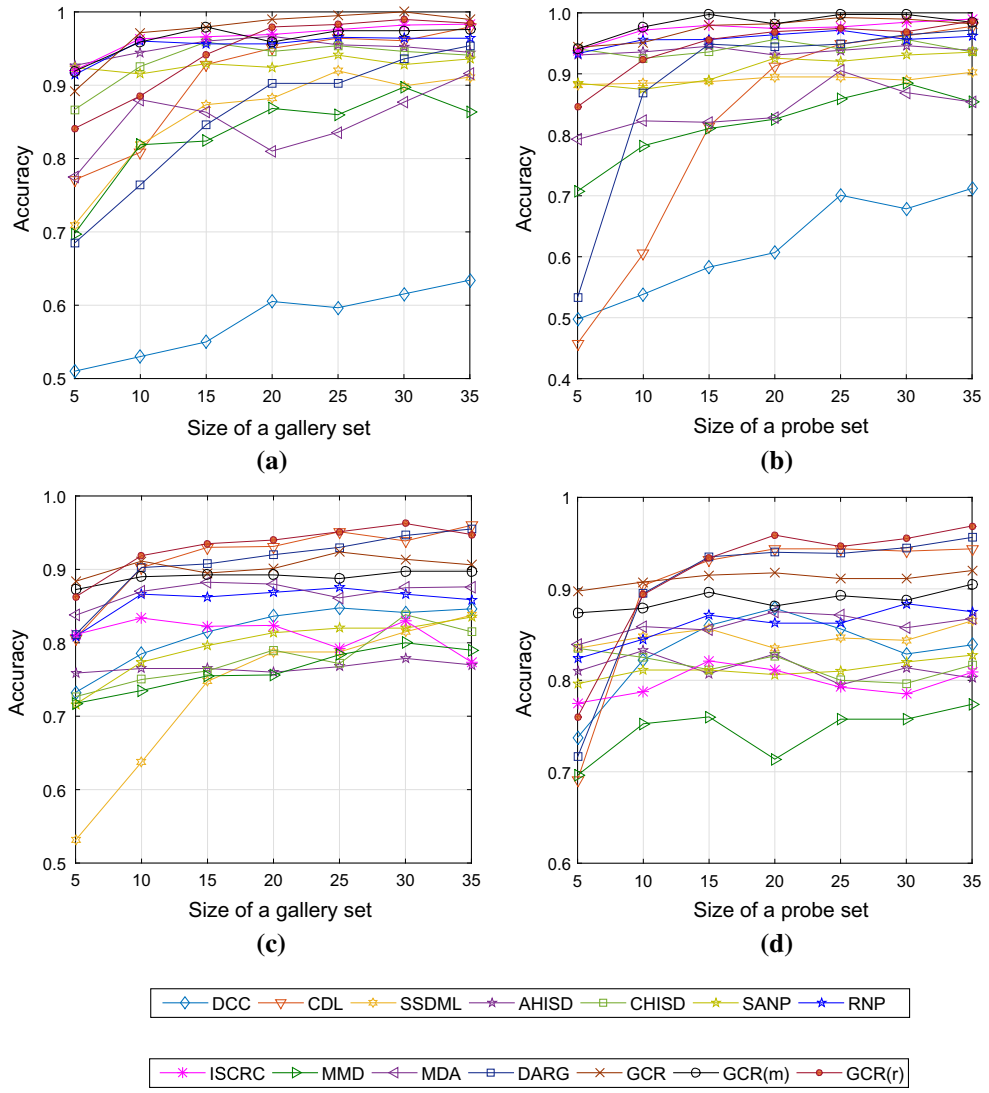


**Fig. 8** The influence of gallery set size on the **a** Honda (size of a probe set = 50) and **c** ETH datasets (size of a probe set = 41), and the influence of probe set size on the **b** Honda and (size of a gallery set = 50), **d** ETH datasets (size of a gallery set = 41)

sification performance. This result supports our advocacy for the SSsR model as providing robust and compact descriptions of probe sets.

Figure 7 shows the running times of fourteen algorithms on the full-sized Mobo dataset. All the methods were implemented in MATLAB and executed on a 4 GHz quad-core machine. GCR and GCR(m) are more efficient than the other methods in both the training and testing phases, with the exception of DCC. However, GCR and GCR(m) have superior prediction accuracy compared to DCC: 98.61% and 98.19%,in comparison to 91.25%.

The single-model methods (SANP, CHISD and AHSID) do not have a training phase. However, they require more time during the testing phase because they use a nearest-neighbor scheme that requires forming a one-to-one set matching. Compared with the multi-model methods (MDA, MMD and DARG), GCR-based methods cost less time in both the training and testing phases. The main reason for this disparity is that these multi-model methods require additional time to construct kernels or extract structure.

ISCRC, like GCR, is a collaborative representation method. ISCRC has comparable training time, however it requires more time during testing, because it has to calculate each pair of probe-gallery sets' distance during testing. Our proposed GCR model instead uses a robust and compact representation (SSsR) for the probe set itself. For the same reason, GCR is faster than CDL and RNP during the testing phase.

## 5.6 Representing Low Cardinality Image Sets

In real applications, each image set may comprise only a small number of images. For example, in the task of multi-

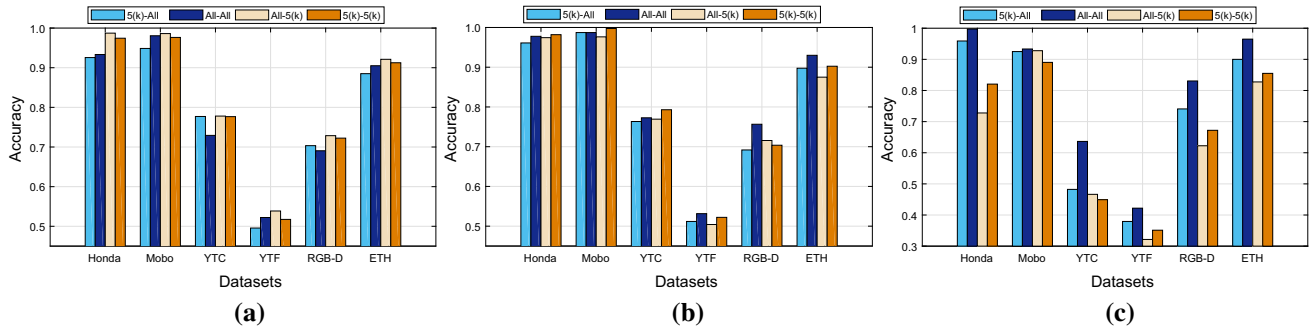


**Fig. 9** The effect of image set size on **a** GCR, **b** GCR(m) and **c** GCR(r) for six datasets. All means that all images are used in each gallery or probe set, and 5(k) means five key images are generated in each gallery or probe set

view object detection, it is often the case that images of each object are available only from a few vantage points. To investigate the efficacy of our proposed GCR model in handling image sets with a small number of images, we compared its performance on the Honda and ETH datasets against the baseline methods as the size of both the gallery and probe sets are varied. In this experiment, five subsets are randomly extracted from the gallery and probe sets except for those sets containing only 5 images; in the latter case, we use PSsR to represent the probe sets.

As shown in Fig. 8, classifiers built on top of the GCR representations using ridge regression and kernel ridge regression with the mean kernel (GCR and GCR(m)) outperform the other methods in most cases, especially when there are only five images in each gallery and probe set. The Riemannian kernel performs poorly because it requires an accurate estimate of the covariance matrix, which is difficult to acquire with a small number of images per set. Similarly, the low cardinality of the image sets contributes to the failure of structure-based methods that attempt to identify subspaces (as in DCC, MMD, and MDA) or capture structure via covariance matrices (as in CDL and DARG).

These results demonstrate that the GCR model has the ability to represent image sets containing only a few images. The main reason for this ability is that the PSsR representation is used in training, so every image in each set is represented collaboratively and used as inputs when training the classifier; accordingly, the learning process makes full use of all available training images. Then at test time, the SSsR model represents each image set stably and collaboratively, so the paucity of images does not unduly affect the representation of the probe sets.

On the other hand, in real application, one image set may contain a few key images and a large number of informative or uninformative images. To simulate this situation, we take advantage of the subspaces identified by GCR to extract five key images from each set. More specifically, the subspace means of each gallery set are taken as key images. For each probe set, the new representations obtained by SSsR are

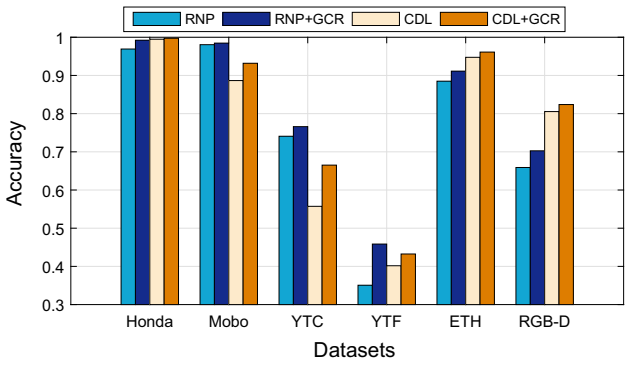


**Fig. 10** Performance of RNP and CDL with and without GCR representations

taken as key images. Four experiments are designed, 5(k)–All, All–All, All–5(k) and 5(k)–5(k), where 5(k) indicates five key images are used in each gallery or probe set, All indicates that all images are used in each gallery or probe set. The results of our proposed GCR framework with different classifiers (linear ridge regression (GCR), mean and Riemannian kernelized version (GCR(m) and GCR(r))) are shown in Fig. 9.

As expected, GCR achieves the best performance in the All-5(k) setting (Fig. 9a). It is reasonable because training GCR with all gallery images is helpful to keep the information as much as possible. Meanwhile, only key images of a probe set can enhance the robustness of voting process. From Fig. 9b, we can see that GCR(m) works better with 5(k)–5(k) on face datasets (Honda, Mobo, YTC and YTF). The reason is that the key images extracted by the subspace means have the similar structure to the mean kernel. But the object image set in RGB-D and ETH contain images captured from different views, which leads to a big variance in each set and hard to represent each set with a limited number of key images. GCR(r) benefits from All–All setting (as shown in Fig. 9c) because all images are sufficient to generate the covariance matrix of each set.

**Table 10** Improvements of the set-based classification methods RNP and CDL with the aid of GCR on six datasets

| | Honda (%) | Mobo (%) | YTC (%) | YTF (%) | ETH (%) | RGB-D (%) |
|---|---|---|---|---|---|---|
| RNP + GCR | 2.38 | 0.42 | 3.40 | 30.74 | 2.97 | 6.63 |
| CDL + GCR | 0.25 | 4.87 | 19.36 | 7.67 | 1.54 | 2.28 |

**Table 11** Classification accuracy of the set-based classification methods RNP and CDL and the traditional classification methods with the aid of GCR on six datasets

| | Honda | Mobo | YTC | YTF | ETH | RGB-D |
|---|---|---|---|---|---|---|
| RNP + GCR | 99.23 | 98.47 | 76.60 | 45.85 | 91.13 | 70.27 |
| CDL + GCR | 99.74 | 93.19 | 66.52 | 43.25 | 96.12 | 82.38 |
| GCR(*) | 99.74 | 98.61 | 79.30 | 53.86 | 96.50 | 83.06 |

GCR(*) denotes the best of RR + GCR, KRR(m) + GCR and KRR(r) + GCR

### 5.7 Combining GCR with Existing Set-Based Classification Methods

GCR provides a representation of each image in each gallery set, and each subspace in each probe set, so after learning the PSsR and SSsR, each image set remains a set, but one whose elements are collaborative representations. Accordingly, it is straight-forward to use GCR in conjunction with existing set-based classification methods.

We take two set-based classification methods, RNP and CDL, as examples. Because RNP and CDL representations are built on statistics of the image set such as the covariance matrix, the probe set is also represented via PSsR rather than SSsR. The classification accuracies and corresponding performance improvements are shown in Fig. 10 and Table 10, respectively. Clearly, these set-based classification methods benefit from our new representation model, especially on the difficult data sets YTC and YTF.

Meanwhile, we compared the performance of RNP + GCR, CDL+ GCR and GCR(*) (the best result of RR + GCR, KRR(m) + GCR and KRR(r) + GCR) in Table 11. The results indicate that, when using GCR, traditional classification methods can outperform complicated set-based classification methods. The main reason for this is that probe sets are well-represented using the SSsR model.

## 6 Conclusion

In this paper, we have proposed a group collaborative representation (GCR) framework for representing set-based image data. GCR consists of two parts: a point-to-set representation (PSsR) that encodes gallery sets by coding each gallery image using all the gallery sets, and a set-to-set representation (SSsR) that encodes probe sets by coding several representative subspaces of the probe set in terms of all the gallery sets. Compared with existing set-based methods, GCR effectively captures set structure information and reduces information loss. In particular, classifiers trained on PSsR representations use all the available data during training, which is important for handling low cardinality gallery sets; and applying these classifiers to the more compact SSsR representations increases the efficiency of the prediction process while also increasing the robustness of the set representation. A series of experiments on six real world data sets have shown that GCR consistently outperforms existing methods on the tasks of set-based face recognition and object categorization.

**Acknowledgements** Funding was provided by National Natural Science Foundation of China (Grant Nos. 61632004, 61773050) and Opening Project of Beijing Key Lab of Traffic Data Analysis and Mining (Grant No. BKLTDAM2017001).

## Appendix 1: Theoretical Proof

In this appendix, we prove the two properties of the proposed PSsR representation that are described in Theorem 1 and Theorem 2 respectively. It can be seen that both theorems provide bounds that quantify these expected behaviors by directly relating the stability properties of the PSsR representation to those of the ridge regression representation.

Actually, when $\lambda_1 = 0$, the PSsR representation of an image $\boldsymbol{x}$ reduces to ridge regression:

$$\boldsymbol{z}^\star = \text{argmin}_z \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{z}\|_2^2 + \lambda_2 \|\boldsymbol{z}\|_2^2.$$

For brevity we will refer to the solution to this problem as $\beta_{RR}(\boldsymbol{x}; \lambda_2)$; straightforward calculations give the closed-form expression

$$\boldsymbol{\beta}_{RR}(\boldsymbol{x}; \lambda_2) = (\boldsymbol{D}^T \boldsymbol{D} + \lambda_2 \boldsymbol{I})^{-1} \boldsymbol{D}^T \boldsymbol{x}.$$

Our main tool in proving these results is the following characterization of the PSsR representation.

**Lemma 1** *Consider the PSsR representation of image* $\boldsymbol{x}$*, given by*

$$\boldsymbol{z}^\star = \text{argmin}_z \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{z}\|_2^2 + \lambda_1 \sum_{j=1}^{g} w_j \|\boldsymbol{z}_{G_j}\|_2 + \lambda_2 \|\boldsymbol{z}\|_2^2.$$

*Define*

$$\tilde{\boldsymbol{D}} = \boldsymbol{D}^T\boldsymbol{D} + \lambda_2\boldsymbol{I}$$
$$\tilde{\boldsymbol{x}} = \tilde{D}^{-1/2}\boldsymbol{D}^T\boldsymbol{x}, \text{ and}$$
$$S = \left\{\boldsymbol{v} \,:\, \|(\tilde{\boldsymbol{D}}^{1/2}\boldsymbol{v})_j\|_2 \le \lambda_1 w_j \text{ for } j = 1,\ldots,g\right\}.$$

*The PSsR representation satisfies*

$$\boldsymbol{z}^\star = \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2) - \tilde{\boldsymbol{D}}^{-1/2}P_S(\tilde{\boldsymbol{x}})),$$

*where $P_S(\tilde{\boldsymbol{x}})$ denotes the projection of $\tilde{\boldsymbol{x}}$ onto the convex set $S$.*

***Proof*** The optimization problem defining $\boldsymbol{z}^\star$ is equivalent to

$$\min_{\substack{z\\ \|z_{G_j}\|_2\le\nu_j}} \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{z}\|_2^2 + \lambda_1\sum_{j=1}^{g} w_j\nu_j + \lambda_2\|\boldsymbol{z}\|_2^2.$$

It is clear that $\nu_j^\star = \|z^\star_{G_j}\|_2$. Furthermore, there are strictly feasible points, so by Slater's condition, strong duality holds. The claimed characterization of $\boldsymbol{z}^\star$ follows from identifying the constraints on the dual optimal point.

To find the Lagrangian function, we observe that the constraints require that $(z_{G_j}, \nu_j)$ be in the Lorentz cone, which is self-dual, so the associated dual variables $(\boldsymbol{\beta}_{G_j}, \gamma_j)$ are also in the Lorentz cone, and the Lagrangian is

$$L(\boldsymbol{z},\boldsymbol{\nu},\boldsymbol{\beta},\boldsymbol{\gamma}) = \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{z}\|_2^2 + \lambda_1\sum_{j=1}^{g} w_j\nu_j + \lambda_2\|\boldsymbol{z}\|_2^2 - \sum_{j=1}^{g}\begin{pmatrix} z_{G_j}\\ \nu_j\end{pmatrix}^T\begin{pmatrix}\boldsymbol{\beta}_{G_j}\\ \gamma_j\end{pmatrix}.$$

Primal optimality requires

$$\nabla_{z_{G_j}} L = (\boldsymbol{D}^T\boldsymbol{D} - 2\lambda_2\boldsymbol{I})\boldsymbol{z} - \boldsymbol{D}^T\boldsymbol{x} - \boldsymbol{\beta} = \boldsymbol{0} \text{ and}$$
$$\nabla_{\nu_j} L = \lambda_1\boldsymbol{w} - \boldsymbol{\gamma} = \boldsymbol{0}.$$

It follows that the Lagrangian dual optimization problem is equivalent to

$$\min_{\|\boldsymbol{\beta}_j\|_2\le\lambda_1 w_j} (\boldsymbol{\beta}+\boldsymbol{D}^T\boldsymbol{x})\tilde{\boldsymbol{D}}^{-1}(\boldsymbol{\beta}+\boldsymbol{D}^T\boldsymbol{x}) = \min_{\|\boldsymbol{\beta}_j\|_2\le\lambda_1 w_j} \|\tilde{\boldsymbol{D}}^{-1/2}\boldsymbol{\beta} + \tilde{\boldsymbol{x}}\|_2^2.$$

Define $\tilde{\boldsymbol{\beta}} = -\tilde{\boldsymbol{D}}^{-1/2}\boldsymbol{\beta}$, then this optimization problem is equivalent to

$$\min_{\|(\tilde{\boldsymbol{D}}^{1/2}\tilde{\boldsymbol{\beta}})_j\|_2\le\lambda_1 w_j} \|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{\beta}}\|_2^2.$$

The minimizer of this problem is the projection of $\tilde{\boldsymbol{x}}$ onto the constraint set $S$; it follows that the dual optimal variable is $\boldsymbol{\beta}^\star = -\tilde{\boldsymbol{D}}^{1/2}P_S(\tilde{\boldsymbol{x}})$, and by the primal optimality condition, the corresponding optimal primal variable is

$$\begin{aligned}\boldsymbol{z}^\star &= (\boldsymbol{D}^T\boldsymbol{D} + \lambda_2\boldsymbol{I})^{-1}(\boldsymbol{\beta}^\star + \boldsymbol{D}^T\boldsymbol{x})\\ &= \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2) - \tilde{\boldsymbol{D}}^{-1/2}P_S(\tilde{\boldsymbol{x}})\end{aligned}$$

as claimed. □

Our first stability result in Theorem 1 quantifies the sense in which similar images have similar PSsR representations.

**Theorem 1** *The PSsR representations of any two images $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ satisfy*

$$\|z_1 - z_2\|_2 \le \|\boldsymbol{\beta}_{RR}(\boldsymbol{x}_1;\lambda_2) - \boldsymbol{\beta}_{RR}(\boldsymbol{x}_2;\lambda_2)\|_2 + 2\frac{\lambda_1}{\lambda_2}.$$

***Proof*** Let $\boldsymbol{w}^i$ denote the group sparsity weights associated with image $\boldsymbol{x}_i$ for $i = 1, 2$. Using the notation of Lemma 1, let

$$S_1 = \left\{\boldsymbol{v} \,:\, \|(\tilde{\boldsymbol{D}}^{1/2}\boldsymbol{v})_j\|_2 \le \lambda_1 w_j^1 \text{ for } j = 1,\ldots,g\right\} \text{ and}$$
$$S_2 = \left\{\boldsymbol{v} \,:\, \|(\tilde{\boldsymbol{D}}^{1/2}\boldsymbol{v})_j\|_2 \le \lambda_1 w_j^2 \text{ for } j = 1,\ldots,g\right\}.$$

By Lemma 1,

$$\begin{aligned}\|z_1 - z_2\|_2 \le{}& \|\boldsymbol{\beta}_{RR}(\boldsymbol{x}_1;\lambda_2) - \boldsymbol{\beta}_{RR}(\boldsymbol{x}_2;\lambda_2)\|_2\\ &+ \|\tilde{\boldsymbol{D}}^{-1}\|_2\|\tilde{\boldsymbol{D}}^{1/2}P_{S_1}(\tilde{\boldsymbol{x}}_1) - \tilde{\boldsymbol{D}}^{1/2}P_{S_2}(\tilde{\boldsymbol{x}}_2)\|_2\end{aligned}$$

The definition of $S_1$ implies that

$$\begin{aligned}\|\tilde{\boldsymbol{D}}^{1/2}P_{S_1}(\tilde{\boldsymbol{x}}_1)\|_2^2 &= \sum\nolimits_{j=1}^{g}\left\|(\tilde{\boldsymbol{D}}^{1/2}P_{S_1}(\tilde{\boldsymbol{x}}_1))_j\right\|_2^2\\ &\le \lambda_1^2\|\boldsymbol{w}^1\|_2^2 \le \lambda_1^2,\end{aligned}$$

where the final inequality follows from the fact that $\boldsymbol{w}^1$ is a probability distribution, so has $\ell_2$-norm at most one; similarly, $\|\tilde{\boldsymbol{D}}^{1/2}P_{S_2}(\tilde{\boldsymbol{x}}_2)\|_2 \le \lambda_1$.

Observe also that

$$\|\tilde{\boldsymbol{D}}^{-1}\|_2 = \lambda_{\min}(\boldsymbol{D}^T\boldsymbol{D} + \lambda_2\boldsymbol{I})^{-1} \le \lambda_{\min}(\lambda_2\boldsymbol{I})^{-1} = \lambda_2^{-1}.$$

The desired bound on $\|z_1 - z_2\|_2$ follows by combining these pieces:

$$\begin{aligned}\|z_1 - z_2\|_2 &\le \|\boldsymbol{\beta}_{RR}(\boldsymbol{x}_1;\lambda_2) - \boldsymbol{\beta}_{RR}(\boldsymbol{x}_2;\lambda_2)\|_2\\ &\quad + \|\tilde{\boldsymbol{D}}^{-1}\|_2\left(\left\|\tilde{\boldsymbol{D}}^{1/2}P_{S_1}(\tilde{\boldsymbol{x}}_1)\right\|_2 + \left\|\tilde{\boldsymbol{D}}^{1/2}P_{S_2}(\tilde{\boldsymbol{x}}_2)\right\|_2\right)\\ &\le \|\boldsymbol{\beta}_{RR}(\boldsymbol{x}_1;\lambda_2) - \boldsymbol{\beta}_{RR}(\boldsymbol{x}_2;\lambda_2)\|_2 + 2\frac{\lambda_1}{\lambda_2}.\end{aligned}$$

□

Before stating our second stability result, we present a useful technical lemma.

**Lemma 2** *Let $\boldsymbol{R}$ have orthonormal rows, so $\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$, and let $\boldsymbol{M}$ be a positive semidefinite matrix. Then for any vector $\boldsymbol{x}$,*

$$\|\boldsymbol{R}\boldsymbol{M}\boldsymbol{x}\|_2 \geq \lambda_{min}(\boldsymbol{M})\|\boldsymbol{R}\boldsymbol{x}\|_2.$$

***Proof*** Since $\boldsymbol{M} \succeq \lambda_{\min}(\boldsymbol{M})\boldsymbol{I}$ and conjugation preserves the semidefinite order, $\boldsymbol{R}\boldsymbol{M}\boldsymbol{R}^T \succeq \lambda_{\min}(\boldsymbol{M})\boldsymbol{I}$. It follows that the smallest eigenvalue of $(\boldsymbol{R}\boldsymbol{M}\boldsymbol{R}^T)^2$ is no smaller than $\lambda_{\min}(\boldsymbol{M})^2$, so

$$\boldsymbol{R}\boldsymbol{M}\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{M}\boldsymbol{R}^T \succeq \lambda_{\min}(\boldsymbol{M})^2\boldsymbol{I}.$$

Conjugating both sides by $\boldsymbol{R}$, we observe that

$$\boldsymbol{P}_{\boldsymbol{R}^T}\boldsymbol{M}\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{M}\boldsymbol{P}_{\boldsymbol{R}^T} \succeq \lambda_{\min}(\boldsymbol{M})^2\boldsymbol{R}^T\boldsymbol{R},$$

where $\boldsymbol{P}_{\boldsymbol{R}^T} = \boldsymbol{R}^T\boldsymbol{R}$ is the orthogonal projector onto the row space of $\boldsymbol{R}$. We therefore conclude that $\boldsymbol{M}\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{M} \succeq \lambda_{\min}(\boldsymbol{M})^2\boldsymbol{R}^T\boldsymbol{R}$, which implies that

$$\begin{aligned}\|\boldsymbol{R}\boldsymbol{M}\boldsymbol{x}\|_2^2 &= \boldsymbol{x}^T\boldsymbol{M}\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{M}\boldsymbol{x} \geq \lambda_{\min}(\boldsymbol{M})^2\boldsymbol{x}^T\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{x} \\ &= \lambda_{\min}(\boldsymbol{M})^2\|\boldsymbol{R}\boldsymbol{x}\|_2^2.\end{aligned}$$

□

Our second stability result in Theorem 2 quantifies the extent to which similar galleries induce similar coordinates in the PSsR representation of a single image.

**Theorem 2** *Let $\boldsymbol{z}^\star$ denote the PSsR representation of an image $\boldsymbol{x}$. Each pair $(z^\star_{G_j}, z^\star_{G_k})$ of subgroups of coordinates satisfies*

$$\begin{aligned}\|z^\star_{G_j} - z^\star_{G_k}\|_2 \leq \|\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_j} - \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_k}\|_2 \\ + \frac{\lambda_1}{\lambda_2}(w_j + w_k).\end{aligned}$$

***Proof*** Let $\boldsymbol{R}_j$ denote the matrix that maps a vector $z$ to $z_{G_j}$ and, similarly, let $\boldsymbol{R}_k$ denote the matrix that maps $z$ to $z_{G_k}$. By Lemma 1, $\boldsymbol{z}^\star = \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2) - \tilde{\boldsymbol{D}}^{-1/2}P_S(\tilde{\boldsymbol{x}})$, so

$$\begin{aligned}\|z^\star_{G_j} - z^\star_{G_k}\|_2 \leq \|\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_j} - \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_k}\|_2 \\ + \|(\boldsymbol{R}_j - \boldsymbol{R}_k)\tilde{\boldsymbol{D}}^{-1/2}P_S(\tilde{\boldsymbol{x}})\|_2.\end{aligned}$$

The matrix $\boldsymbol{R}_j$ has orthonormal rows, so by Lemma 2,

$$\begin{aligned}&\|\boldsymbol{R}_j\tilde{\boldsymbol{D}}^{-1/2}P_S(\tilde{\boldsymbol{x}})\|_2 = \|\boldsymbol{R}_j\tilde{\boldsymbol{D}}^{-1}\tilde{\boldsymbol{D}}^{1/2}P_S(\tilde{\boldsymbol{x}})\|_2 \\ &\leq \|\tilde{\boldsymbol{D}}^{-1}\|_2\|(\tilde{\boldsymbol{D}}^{1/2}P_S(\tilde{\boldsymbol{x}}))_{G_j}\|_2 \leq \frac{\lambda_1}{\lambda_2}w_j.\end{aligned}$$

The last inequality holds because of the definition of $S$ and the observation that

$$\|\tilde{\boldsymbol{D}}^{-1}\|_2 = \lambda_{\min}(\boldsymbol{D}^T\boldsymbol{D} + \lambda_2\boldsymbol{I})^{-1} \leq \lambda_2^{-1}.$$

A similar argument shows that

$$\|\boldsymbol{R}_k\tilde{\boldsymbol{D}}^{-1/2}P_S(\tilde{\boldsymbol{x}})\|_2 \leq \frac{\lambda_1}{\lambda_2}w_k.$$

From these estimates, we conclude that

$$\begin{aligned}\|z^\star_{G_j} - z^\star_{G_k}\|_2 = \|\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_j} - \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_k}\|_2 \\ + \frac{\lambda_1}{\lambda_2}(w_j + w_k)\end{aligned}$$

as claimed. □

The previous bound relates the stability of the PSsR representation to that of the ridge regression representation; for completeness, we use standard arguments to estimate the stability of the ridge regression representation, resulting in a direct relationship between the similarity of $\boldsymbol{D}_j$ and $\boldsymbol{D}_k$ and that of $z_{G_j}$ and $z_{G_k}$.

**Corollary 1** *Let $\boldsymbol{z}^\star$ denote the PSsR representation of an image $\boldsymbol{x}$ and $\boldsymbol{r} = \boldsymbol{x} - \boldsymbol{D}\beta_{RR}(\boldsymbol{x};\lambda_2)$ denote the residual of the ridge regression representation. Each pair $(z^\star_{G_j}, z^\star_{G_k})$ of subgroups of coordinates satisfies*

$$\|z^\star_{G_j} - z^\star_{G_k}\|_2 \leq \frac{1}{\lambda_2}\|\boldsymbol{D}_j - \boldsymbol{D}_k\|_2\|\boldsymbol{r}\|_2 + \frac{\lambda_1}{\lambda_2}(w_j + w_k).$$

***Proof*** The ridge regression coordinates are the solution to the smooth unconstrained optimization problem

$$\text{argmin}_z\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{z}\|_2^2 + \lambda_2\|z\|_2^2,$$

so they are characterized by the property that the gradient of the objective vanishes at $\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)$:

$$(\boldsymbol{D}^T\boldsymbol{D} + \lambda_2\boldsymbol{I})\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2) - \boldsymbol{D}^T\boldsymbol{x} = \boldsymbol{0}.$$

In particular, by considering the appropriate blocks of coordinates in this gradient, we see that

$$\begin{aligned}&\boldsymbol{D}_j^T\boldsymbol{D}\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2) + \lambda_2\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_j} - \boldsymbol{D}_j^T\boldsymbol{x} = \boldsymbol{0} \text{ and} \\ &\boldsymbol{D}_k^T\boldsymbol{D}\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2) + \lambda_2\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_k} - \boldsymbol{D}_k^T\boldsymbol{x} = \boldsymbol{0}.\end{aligned}$$

It follows that

$$\begin{aligned}&\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_j} - \boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_k} \\ &= \frac{1}{\lambda_2}(\boldsymbol{D}_j - \boldsymbol{D}_k)^T(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)),\end{aligned}$$

so

$$\|\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_j}-\boldsymbol{\beta}_{RR}(\boldsymbol{x};\lambda_2)_{G_k}\|_2 \le \frac{1}{\lambda_2}\|\boldsymbol{D}_j-\boldsymbol{D}_k\|_2\|\boldsymbol{r}\|_2.$$

Using this estimate in Lemma 2 gives the desired result. □

## Appendix 2: Optimization Procedures

### Optimization for PSsR

The optimization problem (8) for PSsR is convex and can be solved by various methods. For simplicity in dealing with the group lasso penalty, we employ the alternating direction multiplier method (ADMM) (Boyd et al. 2011) to find the optimal solution. To do so, we introduce an auxiliary vector $\boldsymbol{a} \in \mathbb{R}^{cg}$ of the same size as $z$, and replace (8) with the equivalent problem

$$\min_{z,\boldsymbol{a}} \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{a}\|_2^2+\lambda_1\sum_{j=1}^{g} w_j\|z_{G_j}\|_2+\lambda_2\|z\|_2^2, \quad \text{s.t. } \boldsymbol{a}=z. \tag{18}$$

The augmented Lagrangian function of (18) is

$$\begin{aligned}\mathscr{L}(z,\boldsymbol{a},\boldsymbol{\gamma},\eta) = &\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{a}\|_2^2+\lambda_1\sum_{j=1}^{g} w_j\|z_{G_j}\|_2+\lambda_2\|z\|_2^2\\ &+\langle\boldsymbol{\gamma},z-\boldsymbol{a}\rangle+\frac{\eta}{2}\|z-\boldsymbol{a}\|_2^2,\end{aligned} \tag{19}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^{cg}$ is the Lagrange multiplier, $\eta$ is a positive number that is adaptively updated, and $\langle\cdot,\cdot\rangle$ denotes the inner product between two vectors. The optimization problem (18) can be solved by minimizing (19) by fixing two variables of $(z,\boldsymbol{a},\boldsymbol{\gamma})$, minimizing over the free variable, and alternating over the choice of optimization variable until convergence.

In more detail, to determine $a^{(\tau+1)}$, the value of $\boldsymbol{a}$ at step $\tau+1$, set $z=z^{(\tau)}$, $\boldsymbol{\gamma}=\boldsymbol{\gamma}^{(\tau)}$ and $\eta=\eta^{(\tau)}$ and solve

$$\arg\min_{\boldsymbol{a}} \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{a}\|_2^2+\frac{\eta}{2}\left\|z-\boldsymbol{a}+\frac{\boldsymbol{\gamma}}{\eta}\right\|_2^2. \tag{20}$$

This quadratic program is optimized when its derivative with respect to $\boldsymbol{a}$ is zero, i.e. when

$$\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{a}-\boldsymbol{D}^T\boldsymbol{x}+\eta\boldsymbol{a}-\eta z-\boldsymbol{\gamma}=0,$$

so $\boldsymbol{a}^{(\tau+1)}$ is given by

$$\boldsymbol{a}^{(\tau+1)}=(\boldsymbol{D}^T\boldsymbol{D}+\eta\boldsymbol{I})^{-1}(\boldsymbol{D}^T\boldsymbol{x}+\eta z+\boldsymbol{\gamma}). \tag{21}$$

It is clear that the inverse of $(\boldsymbol{D}^T\boldsymbol{D}+\eta\boldsymbol{I})$ exists as the matrix is symmetric positive definite.

Similarly, when we fix $\boldsymbol{a}=\boldsymbol{a}^{(\tau+1)}$, $\boldsymbol{\gamma}=\boldsymbol{\gamma}^{(\tau)}$ and $\eta=\eta^{(\tau)}$, then $z^{(\tau+1)}$ is determined by solving

$$\arg\min_{z} \lambda_1\sum_{j=1}^{g} w_j\|z_{G_j}\|_2+\lambda_2\|z\|_2^2+\frac{\eta}{2}\left\|z-\left(\boldsymbol{a}-\frac{\boldsymbol{\gamma}}{\eta}\right)\right\|_2^2. \tag{22}$$

It turns out that (22) is equivalent to the following problem,

$$\begin{aligned}\arg\min_{z} &\frac{2\lambda_1}{\sqrt{\eta+2\lambda_2}}\sum_{j=1}^{g} w_j\left\|\sqrt{\eta+2\lambda_2}z_{G_j}\right\|_2\\ &+\left\|\sqrt{\eta+2\lambda_2}z-\frac{\eta}{\sqrt{\eta+2\lambda_2}}\left(\boldsymbol{a}-\frac{\boldsymbol{\gamma}}{\eta}\right)\right\|_2^2.\end{aligned} \tag{23}$$

Using the change of variables $\hat{\boldsymbol{x}}=\frac{\eta}{\sqrt{\eta+2\lambda_2}}(\boldsymbol{a}-\frac{\boldsymbol{\gamma}}{\eta})$ and $\hat{z}=z\sqrt{\eta+2\lambda_2}$, rewrite (23) as

$$\arg\min_{\hat{z}} \frac{1}{2}\|\hat{\boldsymbol{x}}-\hat{z}\|_2^2+\frac{\lambda_1}{\sqrt{\eta+2\lambda_2}}\sum_{j=1}^{g} w_j\|\hat{z}_{G_j}\|_2. \tag{24}$$

Following the analysis in Bach et al. (2012), one can show that

$$\hat{z}_{G_j}=\frac{\hat{\boldsymbol{x}}_{G_j}}{\|\hat{\boldsymbol{x}}_{G_j}\|_2}\max\left(\|\hat{\boldsymbol{x}}_{G_j}\|_2-\frac{w_j\lambda_1}{\sqrt{\eta+2\lambda_2}},0\right),$$

and the update rule for $z$ follows from the definitions of $\hat{\boldsymbol{x}}$ and $\hat{z}$:

$$z_{G_j}^{(\tau+1)}=\frac{\boldsymbol{t}_{G_j}}{\|\boldsymbol{t}_{G_j}\|_2}\max\left(\frac{\|\boldsymbol{t}_{G_j}\|_2-w_j\frac{\lambda_1}{\eta}}{1+\frac{2\lambda_2}{\eta}},0\right) \tag{25}$$

where $\boldsymbol{t}_{G_j}=(\boldsymbol{a}-\frac{\boldsymbol{\gamma}}{\eta})_{G_j}$.

Now let $\boldsymbol{a}=\boldsymbol{a}^{(\tau+1)}$, $z=z^{(\tau+1)}$ and $\eta=\eta^{(\tau)}$. We update the Lagrange multiplier by the amount by which the constraint $\boldsymbol{a}^{(\tau+1)}=z^{(\tau+1)}$ is violated, via

$$\boldsymbol{\gamma}^{(\tau+1)}=\boldsymbol{\gamma}^{(\tau)}+\eta(z-\boldsymbol{a}). \tag{26}$$

The choice of the penalty parameter $\eta$ plays a crucial role in the efficiency of the algorithm. Although in theory a larger value of $\eta$ leads to faster convergence, too large a value can cause numerical difficulties. In general, the correct choice of $\eta$ is problem-dependent. Fortunately, an adaptive updating strategy was proposed in Tao and Yuan (2011); Lin et al. (2011) that dynamically updates $\eta$ via

$$\eta^{(\tau+1)}=\min\{\rho\eta^{(\tau)},\eta_{max}\}. \tag{27}$$

Here $\eta_{max}$ is a given upper bound for $\{\eta^\tau\}$, and $\rho \geq 1$ is a constant. In practice, we set $\eta_{max} = 10^4$ and $\rho = 1.1$.

**Algorithm 1** Solving (18) via ADMM to obtain a PSsR

**Input:** Image $\boldsymbol{x} \in \mathbb{R}^d$, dictionary $\boldsymbol{D} \in \mathbb{R}^{d\times cg}$, weight vector $\boldsymbol{w} \in \mathbb{R}^g$, and parameters $\lambda_1, \lambda_2 \geq 0$.
1: Initialize $z^{(0)} = \boldsymbol{a}^{(0)} = \boldsymbol{\gamma}^{(0)} = 0$, $\eta^{(0)} = 0.1$, $\eta_{max} = 10^4$, $\rho = 1.1$, $\epsilon = 10^{-3}$, $\tau = 0$
2: **while** not converged **do**
3: $\tau \leftarrow \tau + 1$
4: Update $\boldsymbol{a}^{(\tau)}$ using Eq. (21).
5: Update $z^{(\tau)}$ using Eq. (25.)
6: Update the multipliers $\boldsymbol{\gamma}^{(\tau)}$ using Eq. (26).
7: Update $\eta^{(\tau)}$ using Eq.(27).
8: Check the convergence conditions: $\|\boldsymbol{a}^{(\tau)} - z^{(\tau)}\|_\infty < \varepsilon$ and $\max\{\|\boldsymbol{a}^{(\tau)} - \boldsymbol{a}^{(\tau-1)})\|_\infty, \|z^{(\tau)} - z^{(\tau-1)})\|_\infty\} < \varepsilon$.
9: **end while**
**Output:** $z^{(\tau)}, \boldsymbol{a}^{(\tau)}$

Algorithm 1 details the procedure for determining PSsR coordinates using the ADMM method. In each iteration, updating $\boldsymbol{a}$ in (21) requires the construction of $(\boldsymbol{D}^T\boldsymbol{x} + \eta z + \boldsymbol{\gamma})$, which costs $O(dcg)$. After precomputing the eigenvalue decomposition of $\boldsymbol{D}^T\boldsymbol{D}$ in time $O((cg)^3)$, one can apply the inverse of $(\boldsymbol{D}^T\boldsymbol{D} + \eta\boldsymbol{I})$ at each iteration with cost $O((cg)^2)$. Clearly the updates for $z$ and the Lagrange multiplier $\boldsymbol{\gamma}$, (25) and (26), can be computed in time $O(cg)$, the size of $z$. Consequently, assuming a constant number of iterations of ADMM, the complexity of identifying the PSsR representation for an image is $O(dcg + (cg)^2 + (cg)^3)$. Also, the $O((cg)^3)$ cost of the decomposition of $\boldsymbol{D}^T\boldsymbol{D}$ needs only be paid once, then the result can be used in finding the PSsRs of subsequent images. We remark that the computational complexity can be reduced by considering inexact versions of ADMM Ng et al. (2011) or by using suitable surrogate functions Razaviyayn et al. (2013).

## Optimization for SSsR

SSsRs are obtained by solving the optimization problem (9). We find it convenient to rewrite (9) as

$$\begin{aligned}\min_{z,\boldsymbol{y}} &\left\|\begin{bmatrix}\boldsymbol{U}_r^i & -\boldsymbol{D}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}\\ z\end{bmatrix}\right\|_2^2 + \lambda_1\sum_{j=1}^g w_j\left\|\left(\begin{bmatrix}\boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}\\ z\end{bmatrix}\right)_{G_j}\right\|_2\\ &+\lambda_2\left\|\begin{bmatrix}\boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}\\ z\end{bmatrix}\right\|_2^2 + \lambda_3\left\|\begin{bmatrix}\boldsymbol{I} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}\\ z\end{bmatrix}\right\|_2^2\\ &\text{s.t. } \begin{bmatrix}\boldsymbol{1} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}\\ z\end{bmatrix} = 1.\end{aligned} \tag{28}$$

For convenience, define

$$\boldsymbol{a} = \begin{bmatrix}\boldsymbol{y}\\ z\end{bmatrix} \in \mathbb{R}^{m_r^i+cg}$$

$$\boldsymbol{V} = \begin{bmatrix}\boldsymbol{U}_r^i & -\boldsymbol{D}\end{bmatrix} \in \mathbb{R}^{d\times(m_r^i+cg)}$$

$$\boldsymbol{Q} = \begin{bmatrix}\boldsymbol{I} & \boldsymbol{0}\end{bmatrix} \in \mathbb{R}^{m_r^i\times(m_r^i+cg)}, \quad \text{where } \boldsymbol{I} \in \mathbb{R}^{m_r^i\times m_r^i} \text{ and } \boldsymbol{0} \in \mathbb{R}^{m_r^i\times cg}$$

$$\boldsymbol{f} = \begin{bmatrix}\boldsymbol{1}\\ \boldsymbol{0}\end{bmatrix} \in \mathbb{R}^{m_r^i+cg}, \quad \text{where } \boldsymbol{1} \in \mathbb{R}^{m_r^i} \text{ and } \boldsymbol{0} \in \mathbb{R}^{cg}$$

$$\boldsymbol{M} = \begin{bmatrix}\boldsymbol{0} & \boldsymbol{I}\end{bmatrix} \in \mathbb{R}^{cg\times(cg+m_r^i)}, \quad \text{where } \boldsymbol{I} \in \mathbb{R}^{cg\times cg} \text{ and } \boldsymbol{0} \in \mathbb{R}^{cg\times m_r^i}$$

and rewrite (28) as

$$\begin{aligned}&\min_{\boldsymbol{a}} \|\boldsymbol{V}\boldsymbol{a}\|_2^2 + \lambda_1\sum_{j=1}^g w_j\|(\boldsymbol{M}\boldsymbol{a})_{G_j}\|_2 + \lambda_2\|\boldsymbol{M}\boldsymbol{a}\|_2^2 + \lambda_3\|\boldsymbol{Q}\boldsymbol{a}\|_2^2\\ &\text{s.t. } \boldsymbol{f}^T\boldsymbol{a} = 1.\end{aligned} \tag{29}$$

We again opt to solve this optimization problem via ADMM; to do so we introduce an auxiliary variable $\boldsymbol{b}$ to write the equivalent problem

$$\begin{aligned}&\min_{\boldsymbol{a},\boldsymbol{b}} \|\boldsymbol{V}\boldsymbol{a}\|_2^2 + \lambda_1\sum_{j=1}^g w_j\|\boldsymbol{b}_{G_j}\|_2 + \lambda_2\|\boldsymbol{b}\|_2^2 + \lambda_3\|\boldsymbol{Q}\boldsymbol{a}\|_2^2\\ &\text{s.t. } \boldsymbol{f}^T\boldsymbol{a} = 1,\ \boldsymbol{b} = \boldsymbol{M}\boldsymbol{a},\end{aligned} \tag{30}$$

which has the augmented Lagrangian

$$\begin{aligned}&\mathscr{L}(\boldsymbol{a},\boldsymbol{b},\boldsymbol{\gamma}_1,\boldsymbol{\gamma}_2)\\ &= \|\boldsymbol{V}\boldsymbol{a}\|_2^2 + \lambda_1\sum_{j=1}^g w_j\|\boldsymbol{b}_{G_j}\|_2 + \lambda_2\|\boldsymbol{b}\|_2^2 + \lambda_3\|\boldsymbol{Q}\boldsymbol{a}\|_2^2\\ &\quad + \langle\boldsymbol{\gamma}_1, \boldsymbol{b} - \boldsymbol{M}\boldsymbol{a}\rangle + \frac{\eta}{2}\|\boldsymbol{b} - \boldsymbol{M}\boldsymbol{a}\|_2^2 + \langle\boldsymbol{\gamma}_2, \boldsymbol{f}^T\boldsymbol{a} - 1\rangle\\ &\quad + \frac{\eta}{2}|\boldsymbol{f}^T\boldsymbol{a} - 1|^2,\end{aligned} \tag{31}$$

where $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$ are Lagrange multipliers and $\eta$ is a penalty parameter. We find the SSsR by minimizing this augmented Lagrangian with respect to each of the variables $(\boldsymbol{a},\boldsymbol{b},\boldsymbol{\gamma}_1,\boldsymbol{\gamma}_2)$ in an alternating manner.

Variable $\boldsymbol{a}$ is updated by fixing $\boldsymbol{b} = \boldsymbol{b}^{(\tau)}$, $\boldsymbol{\gamma}_1 = \boldsymbol{\gamma}_1^{(\tau)}$, $\boldsymbol{\gamma}_2 = \boldsymbol{\gamma}_2^{(\tau)}$ and $\eta = \eta^{(\tau)}$ and solving

$$\begin{aligned}\arg\min_{\boldsymbol{a}} &\|\boldsymbol{V}\boldsymbol{a}\|_2^2 + \lambda_3\|\boldsymbol{Q}\boldsymbol{a}\|_2^2 + \frac{\eta}{2}\left|\boldsymbol{f}^T\boldsymbol{a} - 1 + \frac{\boldsymbol{\gamma}_2}{\eta}\right|^2\\ &+\frac{\eta}{2}\left\|\boldsymbol{b} - \boldsymbol{M}\boldsymbol{a} + \frac{\boldsymbol{\gamma}_1}{\eta}\right\|_2^2.\end{aligned} \tag{32}$$

The objective is a quadratic in $\boldsymbol{a}$ with minimizer

$$\boldsymbol{a}^{(\tau+1)} = \left(2\boldsymbol{V}^T\boldsymbol{V} + 2\lambda_3\boldsymbol{Q}^T\boldsymbol{Q} + \eta\boldsymbol{f}\boldsymbol{f}^T + \eta\boldsymbol{M}^T\boldsymbol{M}\right)^{-1} \left((\eta-\gamma_2)\boldsymbol{f} + \boldsymbol{M}^T(\eta\boldsymbol{b}+\boldsymbol{\gamma}_1)\right). \tag{33}$$

Note that $\boldsymbol{M}^T\boldsymbol{M} + \boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}$, so the matrix inversion in (33) is well-defined.

The new value for $\boldsymbol{b}$ is obtained by fixing $\boldsymbol{a} = \boldsymbol{a}^{(\tau+1)}$, $\boldsymbol{\gamma}_1 = \boldsymbol{\gamma}_1^{(\tau)}$, $\boldsymbol{\gamma}_2 = \boldsymbol{\gamma}_2^{(\tau)}$ and $\eta = \eta^{(\tau)}$ and optimizing

$$\arg\min_{\boldsymbol{b}} \lambda_1 \sum_{j=1}^{g} w_j \|\boldsymbol{b}_{G_j}\|_2 + \lambda_2\|\boldsymbol{b}\|_2^2 + \frac{\eta}{2}\left\|\boldsymbol{b} - \left(\boldsymbol{M}\boldsymbol{a} - \frac{\boldsymbol{\gamma}_2}{\eta}\right)\right\|_2^2, \tag{34}$$

which has the same form as (22). Applying similar reasoning as leads to (25), we have that

$$\boldsymbol{b}_{G_j}^{(\tau+1)} = \frac{\boldsymbol{t}_{G_j}}{\|\boldsymbol{t}_{G_j}\|}\max\left(\frac{\|\boldsymbol{t}_{G_j}\|_2 - w_j\frac{\lambda_1}{\eta}}{1+\frac{2\lambda_2}{\eta}}, 0\right) \tag{35}$$

where $\boldsymbol{t}_{G_j} = \left(\boldsymbol{M}\boldsymbol{a} - \frac{\boldsymbol{\gamma}_2}{\eta}\right)_{G_j}$.

The Lagrange multipliers $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$, and penalty parameter $\eta$ are updated via

$$\boldsymbol{\gamma}_1^{(\tau+1)} = \boldsymbol{\gamma}_1^{(\tau)} + \eta(\boldsymbol{b} - \boldsymbol{M}\boldsymbol{a}), \tag{36}$$

$$\boldsymbol{\gamma}_2^{(\tau+1)} = \boldsymbol{\gamma}_2^{(\tau)} + \eta(\boldsymbol{f}^T\boldsymbol{a} - 1), \tag{37}$$

and

$$\eta^{(\tau+1)} = \min(\rho\eta^{(\tau)}, \eta_{max}). \tag{38}$$

The complete algorithm for obtaining SSsR coordinates is summarized in Algorithm 2.

**Algorithm 2** Solving (9) via ADMM to obtain an SSsR

**Input:** Images $\mathbf{U} \in \mathbb{R}^{d\times m_r^i}$, dictionary $\boldsymbol{D} \in \mathbb{R}^{d\times cg}$, weight vector $\boldsymbol{w} \in \mathbb{R}^g$, and parameters $\lambda_1, \lambda_2, \lambda_3 \geq 0$.

1: Initialize $\boldsymbol{b}^{(0)} = \boldsymbol{a}^{(0)} = \boldsymbol{\gamma}_1^{(0)} = \boldsymbol{\gamma}_2^{(0)} = \boldsymbol{0}$, $\eta = 0.1$, $\eta_{max} = 10^4$, $\rho = 1.1$, $\varepsilon = 10^{-3}$, $\tau = 0$
2: **while** not converged **do**
3: $\tau \leftarrow \tau + 1$
4: Update $\boldsymbol{a}^{(\tau)}$ using Eq.(33).
5: Update $\boldsymbol{b}^{(\tau)}$ using Eq.(35).
6: Update the multipliers $\boldsymbol{\gamma}_1^{(\tau)}$ using Eq.(36).
7: Update the multipliers $\boldsymbol{\gamma}_2^{(\tau)}$ using Eq.(37).
8: Update penalty parameter $\eta$ using Eq.(38).
9: Check the convergence conditions: $\|\boldsymbol{a}^{(\tau)} - \boldsymbol{z}^{(\tau)}\|_\infty < \varepsilon$ and $\max\{\|\boldsymbol{a}^{(\tau)} - \boldsymbol{a}^{(\tau-1)})\|_\infty, \|\boldsymbol{z}^{(\tau)} - \boldsymbol{z}^{(\tau-1)})\|_\infty\} < \varepsilon$.
10: **end while**

**Output:** $\boldsymbol{a}, \boldsymbol{b}$

## Appendix 3: Impact of Parameters

The proposed GCR models uses four hyperparameters: the number of subspaces $c$ and the trade-off parameters $\lambda_1$, $\lambda_2$, and $\lambda_3$, appearing in (8) and (9).

### Number of Subspaces

As discussed in Sect. 3, the dictionary $\boldsymbol{D}$ is extracted by dividing each gallery set into $c$ subsets and taking the columns of $\boldsymbol{D}$ to be the means of these subsets. The probe sets are also separated into $c$ subsets, each of which is modeled in terms of $\boldsymbol{D}$. Thus, $c$ plays an important role in both GCR models.

To determine the effect of $c$ on the GCR representations, we evaluated the accuracy of ridge regression classifiers fit on six datasets using GCR representations with different values of $c$. Figure 11 shows the accuracy when $c$ is fixed to 5 for the probe set representations and is varied for the gallery set representations, and Fig. 12 shows the accuracy when $c$ is fixed to 5 for the gallery set representations and varied for the probe set representations. The reported accuracies are 10-fold cross-validated.

We see from Fig. 11 that the accuracy is low when $c$ is small for the gallery set representations, and becomes relatively stable when $c$ is in $[3, 10]$. This is reasonable, because when $c$ is small, we are merging together image clusters and losing useful information, but when $c$ is larger, we do not pay any accuracy price for potentially over-partitioning the image clusters. From Fig. 12,

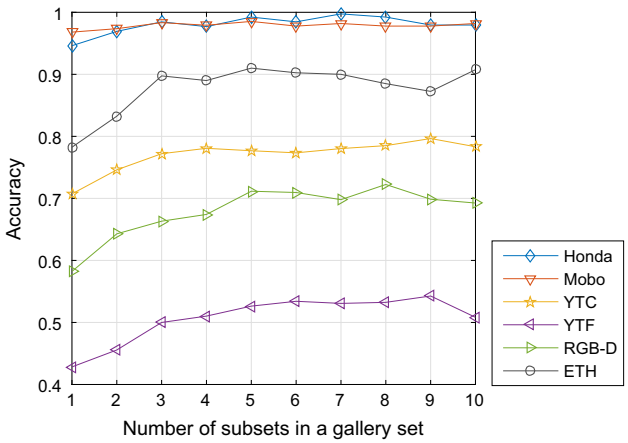


**Fig. 11** Impact of $c$ (the number of subsets in each gallery set) on GCR in terms of classification accuracy where each probe set is represented with five subsets

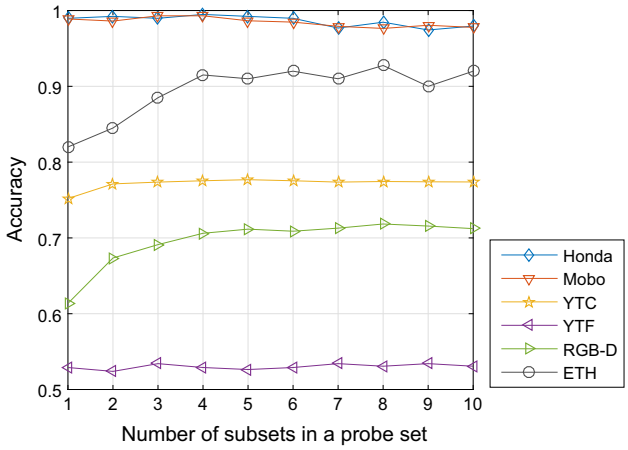


**Fig. 12** Impact of $c$ (the number of subsets in each probe set) on GCR in terms of classification accuracy where each gallery set is represented with five subsets

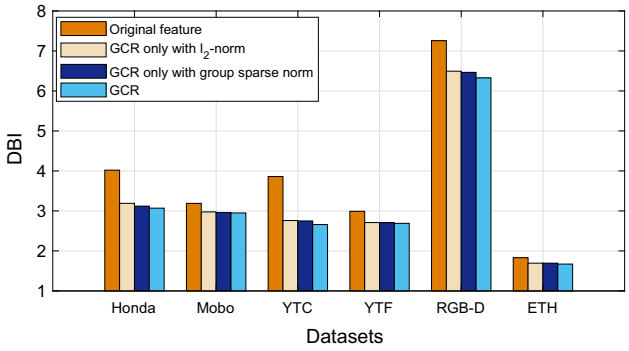


**Fig. 13** The quality of four choices of features used to represent the all six datasets measured in terms of the Davies–Bouldin index (Davies and Bouldin 1979, the smaller the better)

we find that the accuracy increases and then decreases as the number of clusters used to represent the probe sets increases. Here large $c$ means that the probe set is over-segmented relative to the gallery sets, which makes the voting scheme more sensitive to outliers and violates the voting consistency (as discussed in Sect. 4) required in the prediction process. Based on these observations, we set $c$ as 5 for the representations of both gallery and probe sets.

## Trade-off Parameters $\lambda_1$, $\lambda_2$ and $\lambda_3$

There are three terms in the PSsR model (8). In this subsection, we will demonstrate the effect of the two regularizers, the second term ($\ell_2$-norm constraint on new representation) and the third term (group sparsity constraint on new representation).

In the first experiment, we calculate the Davies–Bouldin index (DBI) (Davies and Bouldin 1979) of six datasets for four representations (the original pixel features, PSsR with only $\ell_2$-norm regularization (i.e., $\lambda_1 = 0$), PSsR with only group lasso regularization (i.e., $\lambda_2 = 0$), and PSsR with the full objective). The DBI measure simultaneously evaluates intra-cluster compactness and inter-cluster separation, and is defined as

$$DBI = \frac{1}{L}\sum_{i=1}^{L}\max_{i\neq j}\frac{C_i + C_j}{S_{i,j}} \tag{39}$$

where $C_i = \frac{1}{n_i}\sum_{k=1}^{n_i}\|\boldsymbol{x}_k - \boldsymbol{\mu}_i\|_2$ measures the compactness of the $i$th category and $S_{i,j} = \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|_2$ measures the separation between the $i$th and $j$th categories. Here $L$ is the number of classes, $\boldsymbol{\mu}_i$ is the mean of the images in the $i$th class, and $n_i$ is the number of images in the $i$th class. Lower DBI values indicate that a representation is better able to capture intra- and inter-set relationships.

The DBIs of four representations evaluated on the Honda video training dataset ($L = 20$) are shown in Fig. 13. Clearly the lowest DBI value is obtained by the PSsR, which demonstrates that the collaborative nature of the GCR model as well as the mixture of group sparsity and $l_2$ penalties encourages learning representations that are discriminative and result in compact classes.

From the above results, it can be roughly seen that the parameters $\lambda_1$ and $\lambda_2$ control the trade-offs between group sparsity, ridge regularization, and the reconstruction loss in the PSsR objective (8). To demonstrate the effect of these parameters in detail, we conducted extensive experiments by varying $\lambda_1$ and $\lambda_2$ on six datasets, as shown in Fig. 14. Ridge regression is used and the accuracies reported are 10-fold cross-validated. For simplicity, we set $\lambda_1$ and $\lambda_2$ in the SSsR objective (9) to the same values as those used in the PSsR objective. Since the $\lambda_3$ parameter in the SSsR objective plays a similar role to $\lambda_2$, we take $\lambda_3 = \lambda_2$.

Both parameters took values in $\{10^{-6}, 10^{-5}, \ldots, 10^{0}, 10^{1}\}$. It can be seen that GCR performs stably over a wide range of settings for $\lambda_1$ and $\lambda_2$. In particular, the representations' accuracy is insensitive to $\lambda_2$. Note, however, that large values of $\lambda_1$ ($> 10^{-2}$) tend to result in overly sparse coefficients, which leads to information loss and decreased accuracy. Based on these results, in all other experiments, we set $\lambda_1 = 10^{-3}$, $\lambda_2 = 10^{-1}$ and $\lambda_3 = 10^{-1}$.

The ridge-regression parameter used in fitting the RR and KRR classifiers (10) was tuned among the values $\{10^{-5}, 10^{-4}, \ldots, 10^{3}, 10^{4}\}$. The experimental results show that the classification accuracy is insensitive to $\beta$. Thus we fix $\beta$ in our experiments (with both the RR and KRR classifiers) at $10^{-3}$.

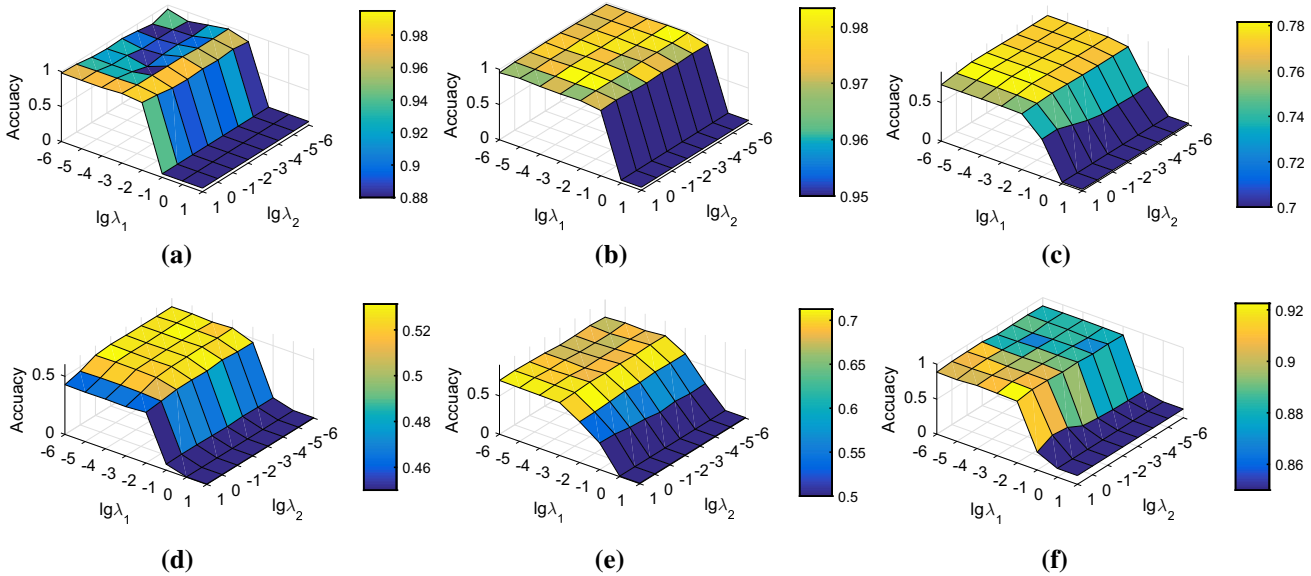


**Fig. 14** Effect of $\lambda_1$ and $\lambda_2$ on GCR in terms of classification accuracy for six datasets **a** Honda, **b** Mobo, **c** YTC, **d** YTF, **e** RGB-D and **f** ETH

## References


Arandjelovic, O., Shakhnarovich, G., Fisher, J., Cipolla, R., & Darrell, T. (2005). Face recognition with image sets using manifold density divergence. In *IEEE conference on computer vision and pattern recognition* (Vol. 1, pp. 581–588). IEEE.

Bach, F., Jenatton, R., Mairal, J., Obozinski, G., et al. (2012). Structured sparsity through convex optimization. *Statistical Science*, *27*(4), 450–468.

Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, *3*(1), 1–122.

Cai, D., He, X., Han, J., & Huang, T. S. (2011). Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(8), 1548–1560.

Cevikalp, H., & Triggs, B. (2010). Face recognition based on image sets. In *IEEE conference on computer vision and pattern recognition* (pp. 2567–2573). IEEE.

Chen, S., Sanderson, C., Harandi, M. T., & Lovell, B. C. (2013). Improved image set classification via joint sparse approximated nearest subspaces. In *IEEE conference on computer vision and pattern recognition* (pp. 452–459). IEEE.

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *1*(2), 224–227.

Deng, W., & Yin, W. (2016). On the global and linear convergence of the generalized alternating direction method of multipliers. *Journal of Scientific Computing*, *66*(3), 889–916.

Gross, R., & Shi, J. (2001). *The CMU motion of body (mobo) database*. Technical report.

Harandi, M. T., Sanderson, C., Shirazi, S., & Lovell, B. C. (2011). Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching. In *IEEE conference on computer vision and pattern recognition* (pp. 2705–2712). IEEE.

Hayat, M., Bennamoun, M., & An, S. (2014). Reverse training: An efficient approach for image set classification. In *European conference on computer vision* (pp. 784–799). Springer.

Hayat, M., Bennamoun, M., & An, S. (2015). Deep reconstruction models for image set classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(4), 713–727.

Hu, Y., Mian, A. S., & Owens, R. (2012). Face recognition using sparse approximated nearest points between image sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(10), 1992–2004.

Huang, Z., Wang, R., Shan, S., & Chen, X. (2014). Learning Euclidean-to-Riemannian metric for point-to-set classification. In *2014 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1677–1684). IEEE.

Kim, M., Kumar, S., Pavlovic, V., & Rowley, H. (2008). Face tracking and recognition with visual constraints in real-world videos. In *IEEE conference on computer vision and pattern recognition* (pp. 1–8). IEEE.

Kim, T. K., Kittler, J., & Cipolla, R. (2007). Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(6), 1005–1018.

Lai, K., Bo, L., Ren, X., & Fox, D. (2011). A large-scale hierarchical multi-view RGB-D object dataset. In *IEEE international conference on robotics and automation* (pp. 1817–1824). IEEE.

Lee, K. C., Ho, J., Yang, M. H., & Kriegman, D. (2003). Video-based face recognition using probabilistic appearance manifolds. In *IEEE conference on computer vision and pattern recognition* (Vol. 1, pp 1–313). IEEE.

Lin, Z., Liu, R., & Su, Z. (2011). Linearized alternating direction method with adaptive penalty for low-rank representation. In *Advances in neural information processing systems* (pp. 612–620).

Lu, C. Y., Min, H., Zhao, Z. Q., Zhu, L., Huang, D. S., & Yan, S. (2012). Robust and efficient subspace segmentation via least squares regression. In *Proceedings of the 12th European conference on computer vision* (pp. 347–360). Springer.

Lu, J., Wang, G., Deng, W., Moulin, P., & Zhou, J. (2015). Multi-manifold deep metric learning for image set classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1137–1145).

Lu, J., Wang, G., & Moulin, P. (2013). Image set classification using holistic multiple order statistics features and localized multi-kernel

metric learning. In *IEEE international conference on computer vision* (pp. 329–336). IEEE.

Mahmood, A., Mian, A., & Owens, R. (2014). Semi-supervised spectral clustering for image set classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 121–128).

Ng, M. K., Wang, F., & Yuan, X. (2011). Inexact alternating direction methods for image recovery. *SIAM Journal on Scientific Computing*, *33*(4), 1643–1668.

Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep face recognition. In *British machine vision conference* (Vol. 1, p. 6).

Razaviyayn, M., Hong, M., & Luo, Z. Q. (2013). A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization*, *23*(2), 1126–1153.

Saunders, C., Gammerman, A., & Vovk, V. (1998). Ridge regression learning algorithm in dual variables. In *Proceedings of the 15th international conference on machine learning* (pp. 515–521). Morgan Kaufmann.

Shakhnarovich, G., Fisher, J. W., & Darrell, T. (2002). Face recognition from long-term observations. In *European conference on computer vision* (pp. 851–865). Springer.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556.

Tao, M., & Yuan, X. (2011). Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM Journal on Optimization*, *21*(1), 57–81.

Uzair, M., Mahmood, A., & Mian, A. (2014). Sparse kernel learning for image set classification. In *Asian conference on computer vision* (pp. 617–631). Springer.

Vidal, R. (2011). Subspace clustering. *IEEE Signal Processing Magazine*, *28*(2), 52–68.

Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, *57*(2), 137–154.

Wang, R., & Chen, X. (2009). Manifold discriminant analysis. In *IEEE conference on computer vision and pattern recognition* (pp. 429–436). IEEE.

Wang, R., Guo, H., Davis, L. S., & Dai, Q. (2012). Covariance discriminative learning: A natural and efficient approach to image set classification. In *IEEE conference on computer vision and pattern recognition* (pp. 2496–2503). IEEE.

Wang, R., Shan, S., Chen, X., & Gao, W. (2008). Manifold–manifold distance with application to face recognition based on image set. In *IEEE conference on computer vision and pattern recognition* (pp. 1–8). IEEE.

Wang, W., Wang, R., Huang, Z., Shan, S., & Chen, X. (2015). Discriminant analysis on Riemannian manifold of Gaussian distributions for face recognition with image sets. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2048–2057).

Wolf, L., Hassner, T., & Maoz, I. (2011). Face recognition in unconstrained videos with matched background similarity. In *IEEE conference on computer vision and pattern recognition* (pp. 529–534). IEEE.

Yang, M., Zhu, P., Van Gool, L., & Zhang, L. (2013). Face recognition based on regularized nearest points between image sets. In *2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG)* (pp. 1–7). IEEE.

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*(1), 49–67.

Zhang, D., Yang, M., & Feng, X. (2011). Sparse representation or collaborative representation: Which helps face recognition? In *Proceedings of the 13th international conference on computer vision* (pp. 471–478). IEEE.

Zhu, P., Zhang, L., Zuo, W., & Zhang, D. (2013). From point to set: Extend the learning of distance metrics. In *2013 IEEE international conference on computer vision (ICCV)* (pp. 2664–2671). IEEE.

Zhu, P., Zuo, W., Zhang, L., & Shiu, S. C. (2014). Image set based collaborative representation for face recognition. *IEEE Transactions on Information Forensics and Security*, *9*(7), 1120–1132.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320.

Zou, H., Zhu, J., Hastie, T., et al. (2008). New multicategory boosting algorithms based on multicategory fisher-consistent losses. *The Annals of Applied Statistics*, *2*(4), 1290–1306.