Transfer Sparse Coding for Robust Image Representation*

Mingsheng Long[†‡], Guiguang Ding[†], Jianmin Wang[†], Jiaguang Sun[†], Yuchen Guo[†], and Philip S. Yu[§]
[†]TNLIST; MOE Lab of Information System Security; School of Software
[‡]Department of Computer Science and Technology, Tsinghua University, Beijing, China
[§]Department of Computer Science, University of Illinois at Chicago, IL, USA
{longmingsheng,guoyc09}@gmail.com {dinggg,jimwang,sunjg}@tsinghua.edu.cn psyu@uic.edu

Abstract

Sparse coding learns a set of basis functions such that each input signal can be well approximated by a linear combination of just a few of the bases. It has attracted increasing interest due to its state-of-the-art performance in BoW based image representation. However, when labeled and unlabeled images are sampled from different distributions, they may be quantized into different visual words of the codebook and encoded with different representations, which may severely degrade classification performance. In this paper, we propose a Transfer Sparse Coding (TSC) approach to construct robust sparse representations for classifying cross-distribution images accurately. Specifically, we aim to minimize the distribution divergence between the labeled and unlabeled images, and incorporate this criterion into the objective function of sparse coding to make the new representations robust to the distribution difference. Experiments show that TSC can significantly outperform state-of-the-art methods on three types of computer vision datasets.

1. Introduction

In computer vision, image representation is a crucial procedure for image processing and understanding. As a powerful tool for finding succinct representations of stimuli and capturing high-level semantics in visual data, *sparse coding* can represent images using only a few active coefficients. This makes the sparse representations easy to interpret and manipulate, and facilitates efficient content-based image indexing and retrieval. Sparse coding is receiving increasing interest in machine learning, pattern recognition, signal processing [9, 11, 8], and has been successfully applied to image classification [22, 12, 24] and face recognition [21, 6].

One major computational problem of sparse coding is to improve the quality of the sparse representation while maximally preserving the signal fidelity. To achieve this goal, many works have been proposed to modify the sparsity constraint. Liu *et al*. [10] added nonnegative constraint to the sparse coefficients. Gao *et al*. [6] introduced a Laplacian term of coefficients in sparse coding, which was extended to an efficient algorithm in Cai *et al*. [24]. Wang *et al*. [20] adopted the weighted ℓ_2-norm for the sparsity constraint. Another line of works focus on improving the signal fidelity, *e.g*., robust sparse coding proposed by Yang *et al*. [23].

However, when labeled and unlabeled images are sampled from *different* distributions, they may be quantized into different visual words of the codebook and encoded with different representations. In this case, the dictionary learned from the labeled images cannot effectively encode the unlabeled images with high fidelity, and also the unlabeled images may reside far away from the labeled images under the new representation. This distribution difference will greatly challenge the robustness of existing sparse coding algorithms for cross-distribution image classification problems.

Recently, the literature has witnessed an increasing focus on *transfer learning* [15] problems where the labeled training data and unlabeled test data are sampled from different probability distributions. This is a very common scenario in real applications, since training and test data are usually collected in different time periods, or under different conditions. In this case, standard classifiers such as SVM and logistic regression trained on the labeled data may fail to make correct predictions on the unlabeled data [13, 14, 16, 17]. To improve the generalization performance of supervised classifiers across different distributions, Pan *et al*. [13, 14] proposed to extract a "good" feature representation through which the probability distributions of labeled and unlabeled data are drawn close. It achieves much better classification performance by explicitly reducing distribution divergence.

*Corresponding author: Jianmin Wang. This work is supported in part by National HGJ Key Project (2010ZX01042-002-002), National High-Tech Development Program (2012AA040911), National Basic Research Program (2009CB320700), and National Natural Science Foundation of China (61073005 and 61271394), and Philip S. Yu is partially supported by US NSF through grants IIS-0905215, CNS-1115234, IIS-0914934, DBI-0960443, and OISE-1129076, US Department of Army through grant W911NF-12-1-0066, Google Mobile 2014 Program and KAU grant.

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.59

Inspired by recent progress in sparse coding and transfer learning, we propose a novel *Transfer Sparse Coding* (TSC) algorithm to construct robust sparse representations for classifying cross-distribution images accurately. We aim to minimize the distribution divergence between labeled and unlabeled images using a nonparametric distance measure. Specifically, we incorporate this criterion into the objective function of sparse coding to make the new representations of the labeled and unlabeled images close to each other. In this way, the induced representations are made robust for cross-distribution image classification problems. Moreover, to enrich the new representations with more discriminating power, we also incorporate the graph Laplacian term of coefficients [24] in our objective function. Extensive experimental results verify the effectiveness of the TSC approach.

2. Related Work

In this section, we discuss prior works that are most related to ours, including sparse coding and transfer learning.

Recently, sparse coding has been a hot research focus in computer vision. To solve the ℓ_1-regularized least squares problem more efficiently, Lee *et al*. [9] proposed a feature-sign search method to reduce the nondifferentiable problem to an unconstrained quadratic programming (QP), which accelerates the optimization process. Our work adapts Lee's method to solve the proposed TSC optimization problem. For adapting the dictionary to achieve sparse representation, Aharon *et al*. [1] proposed a K-SVD method to learn the dictionary using orthogonal matching pursuit or basis pursuit. Our work aims to discover a *shared* dictionary which can encode both labeled and unlabeled data sampled from different probability distributions. To improve the quality of sparse representations, researchers have modified the sparse constraint by adding nonnegative constraint [10], graph regularization [6, 24], weighted ℓ_2-norm constraint [20], etc. Our approach aims to construct robust sparse representations for cross-distribution image classification problems, which is a different learning goal from the previous works.

In the machine learning literature, transfer learning [15], which aims to transfer knowledge between the labeled and unlabeled data sampled from different distributions, has also attracted extensive research interest. To achieve this goal, Pan *et al*. proposed a Transfer Component Analysis (TCA) method to reduce the Maximum Mean Discrepancy (MMD) [7] between the labeled and unlabeled data, and simultaneously minimize the reconstruction error of the input data using PCA. Different from their method, our work focuses on learning robust image representations by building an adaptive model based on sparse coding. Lastly, Quanz *et al*. [16, 17] have explored sparse coding to extract features for knowledge transfer. However, their method adopts a kernel density estimation (KDE) technique to estimate the PDFs of distributions and then minimizes the Jensen-Shannon divergence between them. This is a more restricted procedure than TSC and is prone to overfitting. Moreover, our work additionally incorporates the graph Laplacian term of coefficients [24] in the objective function, which can discover more discriminating representations for classification tasks.

3. Preliminaries

3.1. Sparse Coding

Given a data matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$, with n data points sampled in the m-dimensional feature space, let $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_k] \in \mathbb{R}^{m \times k}$ be the *dictionary* matrix where each column $\mathbf{b}_i$ represents a basis vector in the dictionary, and let $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_n] \in \mathbb{R}^{k \times n}$ be the *coding* matrix where each column $\mathbf{s}_i$ is a sparse representation for a data point $\mathbf{x}_i$. The goal of *sparse coding* is to learn a dictionary (over-complete if $k > m$) and corresponding sparse codes such that input data can be well approximated [16]. Assuming the reconstruction error for a data point follows a zero-mean Gaussian distribution with isotropic covariance, while taking a Laplace prior for the coding coefficients and a uniform prior for the basis vectors, then the maximum a posterior estimate (MAP) of $\mathbf{B}$ and $\mathbf{S}$ given $\mathbf{X}$ is reduced to

$$\min_{\mathbf{B},\mathbf{S}} \|\mathbf{X} - \mathbf{BS}\|_F^2 + \lambda \sum_{i=1}^{n} |\mathbf{s}_i| \quad s.t.\ \|\mathbf{b}_i\|^2 \leq c, \forall i = 1, \ldots, k \tag{1}$$

where λ is a tunable regularization parameter to trade off the sparsity of coding and the approximation of input data. The constraints on the basis vectors are to control the model complexity. Although the objective function in Equation (1) is not convex in both variables, it is convex in either $\mathbf{B}$ or $\mathbf{S}$. Therefore, it can be solved by alternatingly optimizing one variable while fixing the other one. Finally, it can be reduced to an ℓ_1-regularized least squares problem and an ℓ_2-constrained least squares problem, both of which can be solved efficiently by existing optimization software [9, 11].

3.2. Graph Regularized Sparse Coding

To make the basis vectors respect the intrinsic geometric structure underlying the input data, Cai *et al*. [24] proposed a Graph Regularized Sparse Coding (GraphSC) method, which further explores the manifold assumption [2]. GraphSC assumes that if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close in the intrinsic geometry of data distribution, then their codings $\mathbf{s}_i$ and $\mathbf{s}_j$ are also close. Given a set of m-dimensional data points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, GraphSC constructs a p-nearest neighbor graph G with n vertices each representing a data point. Let $\mathbf{W}$ be the weight matrix of G, if $\mathbf{x}_i$ is among the p-nearest neighbor of $\mathbf{x}_j$ or vice versa, $W_{ij} = 1$; otherwise, $W_{ij} = 0$. Define $d_i = \sum_{j=1}^{n} W_{ij}$, $\mathbf{D} = \mathrm{diag}(d_1, \ldots, d_n)$, and graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$. A reasonable criterion for preserving the geometric structure in graph G is to minimize $\frac{1}{2} \sum_{i,j=1}^{n} \|\mathbf{s}_i - \mathbf{s}_j\|^2 W_{ij} = \mathrm{tr}\left(\mathbf{SLS}^{\mathrm{T}}\right)$. Integrating

this criterion into Equation (1) leads to the GraphSC [6, 24]:

$$\min_{\mathbf{B},\mathbf{S}} \|\mathbf{X}-\mathbf{BS}\|_F^2 + \gamma \mathrm{tr}\left(\mathbf{SLS}^{\mathrm{T}}\right) + \lambda \sum\nolimits_{i=1}^{n} |\mathbf{s}_i| \quad s.t. \quad \|\mathbf{b}_i\|^2 \leq c, i = 1, \ldots, k \tag{2}$$

where γ is a graph regularization parameter to trade off the weight between sparse coding and geometric preservation.

4. Transfer Sparse Coding

In this section, we present the Transfer Sparse Coding (TSC) algorithm for robust image representation, which extends GraphSC by taking into account the minimization of distribution divergence between labeled and unlabeled data.

4.1. Problem Definition

Given labeled data $\mathcal{D}_l = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{n_l}, y_{n_l})\}$ with n_l examples, unlabeled data $\mathcal{D}_u = \{\mathbf{x}_{n_l+1}, \ldots, \mathbf{x}_{n_l+n_u}\}$ with n_u examples, denote $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}, n = n_l + n_u$ as the input data matrix. Assume that the labeled and unlabeled data are sampled from *different* probability distributions in an m-dimensional feature space. Frequently used notations and descriptions are summarized in Table 1.

Problem 1 (Transfer Sparse Coding) *Given labeled data $\mathcal{D}_l$ and unlabeled data $\mathcal{D}_u$ under different distributions, our goal is to learn a dictionary $\mathbf{B}$ and a sparse coding $\mathbf{S}$ which performs robustly across the labeled and unlabeled data.*

With Transfer Sparse Coding (TSC), we aim to construct a *robust* representation for images sampled from different distributions. In this way, a supervised classifier trained on the labeled data can generalize better on the unlabeled data.

4.2. Objective Function

To make sparse coding robust to different probability distributions, one may expect that the basis vectors can capture the commonality underlying both the labeled and unlabeled data, rather than only the individual property in the labeled data. However, even in the extracted k-dimensional sparse representation, the distribution difference between labeled and unlabeled data will still be significantly large. Thus one major computational problem is to reduce the distribution difference by explicitly minimizing some predefined distance measures. To realize this idea, a natural strategy is to make the probability distributions of labeled and unlabeled data close to each other in the sparse representation. That is, by representing all data points $\mathbf{X}$ with the learned coding matrix $\mathbf{S}$, the probability distributions of the sparse codes for the labeled and unlabeled data should be close enough. In this paper, we follow [7, 13, 14] and adopt the empirical *Maximum Mean Discrepancy* (MMD) as the nonparametric distance measure to compare different distributions, which

Table 1. Notations and descriptions used in this paper.

Notation	Description	Notation	Description
$\mathcal{D}_l, \mathcal{D}_u$	labeled/unlabeled data	$\mathbf{X}$	input data matrix
n_l, n_u	#labeled/unlabeled examples	$\mathbf{B}$	dictionary matrix
m	#shared features	$\mathbf{S}$	coding matrix
k, p	#basis vectors/nearest neighbors	$\mathbf{M}$	MMD matrix
μ, γ, λ	MMD/graph/sparsity reg. param.	$\mathbf{L}$	graph Laplacian matrix

computes the distance between the sample means of the labeled and unlabeled data in the k-dimensional coefficients:

$$\left\| \frac{1}{n_l} \sum_{i=1}^{n_l} \mathbf{s}_i - \frac{1}{n_u} \sum_{j=n_l+1}^{n_l+n_u} \mathbf{s}_j \right\|^2 = \sum_{i,j=1}^{n} \mathbf{s}_i^{\mathrm{T}} \mathbf{s}_j M_{ij} = \mathrm{tr}\left(\mathbf{SMS}^{\mathrm{T}}\right) \tag{3}$$

where $\mathbf{M}$ is the MMD matrix and is computed as follows

$$M_{ij} = \begin{cases} 1/n_l^2, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_l \\ 1/n_u^2, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_u \\ \frac{-1}{n_l n_u}, & \text{otherwise} \end{cases} \tag{4}$$

By regularizing Equation (2) with Equation (3), dictionary matrix $\mathbf{B}$ is refined and the probability distributions of labeled and unlabeled data are drawn close under the new representation $\mathbf{S}$. We obtain the objective function for TSC:

$$\min_{\mathbf{B},\mathbf{S}} \|\mathbf{X}-\mathbf{BS}\|_F^2 + \mathrm{tr}\left(\mathbf{S}\left(\mu\mathbf{M}+\gamma\mathbf{L}\right)\mathbf{S}^{\mathrm{T}}\right) + \lambda \sum\nolimits_{i=1}^{n} |\mathbf{s}_i| \quad s.t. \quad \|\mathbf{b}_i\|^2 \leq c, \forall i = 1, \ldots, k \tag{5}$$

where $\mu > 0$ is the MMD regularization parameter trading off the weight between GraphSC and distribution matching. To compare the effectiveness between MMD regularization and graph regularization (GraphSC), we refer to the special case of TSC with $\gamma = 0$ as $\mathrm{TSC}_{\mathrm{MMD}}$ and test it empirically.

The MMD regularization in Equation 5 is important to make TSC robust to different probability distributions. According to Gretton *et al.* [7], MMD will asymptotically approach zero if and only if the two distributions are the same. By minimizing MMD, TSC can match distributions between labeled and unlabeled data based on sparse coding.

Following [9, 11, 24], we divide the optimization of TSC into two iterative steps: 1) learning transfer sparse codes $\mathbf{S}$ with dictionary $\mathbf{B}$ fixed, *i.e.*, an ℓ_1-regularized least squares problem; and 2) learning dictionary $\mathbf{B}$ with transfer sparse codes $\mathbf{S}$ fixed, *i.e.*, an ℓ_2-constrained least squares problem.

4.3. Learning Transfer Sparse Codes

We solve optimization problem (5) for transfer sparse codes $\mathbf{S}$. By fixing dictionary $\mathbf{B}$, problem (5) becomes

$$\min_{\mathbf{S}} \|\mathbf{X}-\mathbf{BS}\|_F^2 + \mathrm{tr}\left(\mathbf{S}\left(\mu\mathbf{M}+\gamma\mathbf{L}\right)\mathbf{S}^{\mathrm{T}}\right) + \lambda \sum\nolimits_{i=1}^{n} |\mathbf{s}_i| \tag{7}$$

Unfortunately, problem (7) is nondifferentiable when $\mathbf{s}_i$ takes values of 0, which makes standard unconstrained optimization techniques infeasible. Several recent approaches

Algorithm 1: Learning Transfer Sparse Codes

Input: Data matrix $\mathbf{X}$, dictionary $\mathbf{B}$, MMD matrix $\mathbf{M}$, graph Laplacian matrix $\mathbf{L}$, MMD/graph/sparsity regularization parameters μ, γ, λ.
Output: Current optimal coding matrix $\mathbf{S}^* = [\mathbf{s}_1^*, \ldots, \mathbf{s}_n^*]$.

begin Learning Transfer Sparse Codes
  foreach $\mathbf{s}_i, i \in [1, n]$ **do**
    step Initialize
      $\mathbf{s}_i := \mathbf{0}, \boldsymbol{\theta} := \mathbf{0}$, and active set $\mathcal{A} := \emptyset$, where $\theta_j \in \{-1, 0, 1\}$ denotes $\operatorname{sign}(s_i^{(j)})$.
    step Activate
      From zero coefficients of $\mathbf{s}_i$, select $j := \arg\max_j |\nabla_i^{(j)} g(\mathbf{s}_i)|$. Activate $s_i^{(j)}$ (add j to $\mathcal{A}$) only if it locally improves (9), namely:
        If $\nabla_i^{(j)} g(\mathbf{s}_i) > \lambda$, then set $\theta_j := -1, \mathcal{A} := \{j\} \cup \mathcal{A}$; else if $\nabla_i^{(j)} g(\mathbf{s}_i) < -\lambda$, then set $\theta_j := 1, \mathcal{A} := \{j\} \cup \mathcal{A}$.
    step Feature-sign Search
      Let $\widehat{\mathbf{B}}$ be a submatrix of $\mathbf{B}$ that contains only the columns in $\mathcal{A}$; let $\widehat{\mathbf{s}}_i$, $\widehat{\mathbf{h}}_i$, and $\widehat{\boldsymbol{\theta}}$ be subvectors of $\mathbf{s}_i$, $\mathbf{h}_i$, and $\boldsymbol{\theta}$ in $\mathcal{A}$, respectively.
      Compute the solution to the resulting unconstrained QP: $\min \tilde{f}(\widehat{\mathbf{s}}_i) = \|\mathbf{x}_i - \widehat{\mathbf{B}}\widehat{\mathbf{s}}_i\|^2 + (\mu M_{ii} + \gamma L_{ii})\widehat{\mathbf{s}}_i^{\mathrm{T}}\widehat{\mathbf{s}}_i + \widehat{\mathbf{s}}_i^{\mathrm{T}}\widehat{\mathbf{h}}_i + \lambda\widehat{\boldsymbol{\theta}}^{\mathrm{T}}\widehat{\mathbf{s}}_i$
        Let $\partial\tilde{f}(\widehat{\mathbf{s}}_i)/\partial\widehat{\mathbf{s}}_i := \mathbf{0}$, we can obtain the optimal value of $\mathbf{s}_i$ under the current $\mathcal{A}$:

$$\widehat{\mathbf{s}}_i^{\text{new}} := \left(\widehat{\mathbf{B}}^{\mathrm{T}}\widehat{\mathbf{B}} + (\mu M_{ii} + \gamma L_{ii})\mathbf{I}\right)^{-1}\left(\widehat{\mathbf{B}}^{\mathrm{T}}\mathbf{x}_i - \left(\lambda\widehat{\boldsymbol{\theta}} + \widehat{\mathbf{h}}_i\right)/2\right) \tag{6}$$

      Perform a discrete line search on the closed line segment from $\widehat{\mathbf{s}}_i$ to $\widehat{\mathbf{s}}_i^{\text{new}}$:
        Check the objective value at $\widehat{\mathbf{s}}_i^{\text{new}}$ and all other points where any coefficient changes sign.
        Update $\widehat{\mathbf{s}}_i$ (and the corresponding entries in $\mathbf{s}_i$) to the point with the lowest objective value.
      Remove zero coefficients of $\widehat{\mathbf{s}}_i$ from $\mathcal{A}$ and update $\boldsymbol{\theta} := \operatorname{sign}(\mathbf{s}_i)$.
    step Check Optimality Conditions
      (a) Optimality condition for nonzero coefficients: $\nabla_i^{(j)} g(\mathbf{s}_i) + \lambda\operatorname{sign}(s_i^{(j)}) = 0, \forall s_i^{(j)} \neq 0$
        If condition (a) is not satisfied, go to step "Feature-sign Search" (without any new activation); else check condition (b).
      (b) Optimality condition for zero coefficients: $|\nabla_i^{(j)} g(\mathbf{s}_i)| \leq \lambda, \forall s_i^{(j)} = 0$
        If condition (b) is not satisfied, go to step "Activate"; otherwise return $\mathbf{s}_i$ as the optimal solution, redenote it as $\mathbf{s}_i^*$.

have been proposed to solve the ℓ_1-regularized least squares problem [1, 9, 11, 24], where the *coordinate descent* optimization strategy is often adopted to update each vector $\mathbf{s}_i$ individually with the other vectors $\{\mathbf{s}_j\}_{j \neq i}$ fixed. To facilitate vector-wise manipulations, we rewrite problem (7) as

$$\min_{\{\mathbf{s}_i\}} \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{B}\mathbf{s}_i\|^2 + \sum_{i,j=1}^{n} (\mu M_{ij} + \gamma L_{ij})\mathbf{s}_i^{\mathrm{T}}\mathbf{s}_j + \lambda\sum_{i=1}^{n}|\mathbf{s}_i| \tag{8}$$

The optimization problem involving only $\mathbf{s}_i$ is reduced to

$$\begin{aligned}\min_{\mathbf{s}_i} f(\mathbf{s}_i) = \|\mathbf{x}_i - \mathbf{B}\mathbf{s}_i\|^2 + \lambda\sum\nolimits_{j=1}^{k}|s_i^{(j)}| \\ + (\mu M_{ii} + \gamma L_{ii})\mathbf{s}_i^{\mathrm{T}}\mathbf{s}_i + \mathbf{s}_i^{\mathrm{T}}\mathbf{h}_i\end{aligned} \tag{9}$$

$\mathbf{h}_i = 2\sum_{j \neq i}(\mu M_{ij} + \gamma L_{ij})\mathbf{s}_j$, $s_i^{(j)}$ is jth element of $\mathbf{s}_i$.

We adapt the *feature-sign search* algorithm [9] to solve the optimization problem (9). In nonsmooth optimization methods for solving nondifferentiable problems, a necessary condition for a parameter vector to be a local minimum is that the zero-vector is an element of the subdifferential—the set containing all subgradients at the parameter vector [5]. Define $g(\mathbf{s}_i) = \|\mathbf{x}_i - \mathbf{B}\mathbf{s}_i\|^2 + (\mu K_{ii} + \gamma L_{ii})\mathbf{s}_i^{\mathrm{T}}\mathbf{s}_i + \mathbf{s}_i^{\mathrm{T}}\mathbf{h}_i$, then $f(\mathbf{s}_i) = g(\mathbf{s}_i) + \lambda\sum_{j=1}^{k}|s_i^{(j)}|$. Let $\nabla_i^{(j)}|\mathbf{s}_i|$ be the subdifferentiable value of the jth coefficient of $\mathbf{s}_i$: if $|s_i^{(j)}| > 0$, $\nabla_i^{(j)}|\mathbf{s}_i| = \operatorname{sign}(s_i^{(j)})$; else $s_i^{(j)} = 0$, $\nabla_i^{(j)}|\mathbf{s}_i|$ is nondifferentiable and can take values in $\{-1, 1\}$. The optimality conditions for getting minimum value of $f(\mathbf{s}_i)$ is

$$\begin{cases}\nabla_i^{(j)} g(\mathbf{s}_i) + \lambda\operatorname{sign}(s_i^{(j)}) = 0, & \text{if } |s_i^{(j)}| \neq 0 \\ |\nabla_i^{(j)} g(\mathbf{s}_i)| \leq \lambda, & \text{otherwise}\end{cases} \tag{10}$$

We consider how to select optimal subgradients $\nabla_i^{(j)} f(\mathbf{s}_i)$ when the optimality conditions (10) are violated, that is, $|\nabla_i^{(j)} g(\mathbf{s}_i)| > \lambda$ if $s_i^{(j)} = 0$. Suppose that $\nabla_i^{(j)} g(\mathbf{s}_i) > \lambda$, which implies $\nabla_i^{(j)} f(\mathbf{s}_i) > 0$ regardless of $\operatorname{sign}(s_i^{(j)})$. In this case, to decrease $f(\mathbf{s}_i)$, we need to decrease $s_i^{(j)}$. Since $s_i^{(j)}$ starts at zero, any infinitesimal adjustment to $s_i^{(j)}$ will take it negative. Thus we directly let $\operatorname{sign}(s_i^{(j)}) = -1$. Similarly, if $\nabla_i^{(j)} g(\mathbf{s}_i) < -\lambda$, we directly let $\operatorname{sign}(s_i^{(j)}) = 1$.

Notice that, if we have known the signs of $s_i^{(j)}$'s at the optimal value, we can just replace each term $|s_i^{(j)}|$ with either s_i^j (if $s_i^{(j)} > 0$), $-s_i^j$ (if $s_i^j < 0$), or 0 (if $s_i^j = 0$). Thus by considering only nonzero coefficients, problem (9) is reduced to an unstrained quadratic optimization problem (QP), which can be solved analytically and efficiently. The sketch of learning transfer sparse codes $\{\mathbf{s}_i : i \in [1, n]\}$ is:

- for each $\mathbf{s}_i$, search for the signs of $\{s_i^{(j)} : j \in [1, k]\}$;
- solve the equivalent QP problem to get the optimal $\mathbf{s}_i^*$ that minimizes the vector-wise objective function (9);

- return the optimal coding matrix $\mathbf{S}^* = [\mathbf{s}_i^*, \ldots, \mathbf{s}_n^*]$.

It maintains an *active set* $\mathcal{A} \triangleq \{j | s_i^{(j)} = 0, \nabla_i^{(j)} g(\mathbf{s}_i) > \lambda\}$ for potentially nonzero coefficients and their corresponding signs $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_k]$ while updating each $\mathbf{s}_i$, and systematically searches for the optimal active set and coefficients signs which minimize objective function (9). In each *activate* step, the algorithm uses the zero-value whose violation to the optimality condition $|\nabla_i^{(j)} g(\mathbf{s}_i)| > \lambda$ is the largest. In each *feature-sign* step: 1) given a current value for the active set and the signs, it computes the analytical solution $\mathbf{s}_i^{\text{new}}$ to the resulting unconstrained QP; 2) it updates the solution, the active set, and the signs using an efficient discrete line search between the current solution and $\mathbf{s}_i^{\text{new}}$. The complete learning procedure is summarized in Algorithm 1.

4.4. Learning Dictionary

Learning the dictionary $\mathbf{B}$ with the coding $\mathbf{S}$ fixed is reduced to the following ℓ_2-constrained optimization problem

$$\min_{\mathbf{B}} \|\mathbf{X} - \mathbf{BS}\|_F^2, \quad s.t. \ \|\mathbf{b}_i\|^2 \leq c, \forall i = 1, \ldots, k \quad (11)$$

This problem has been well studied by prior works [9, 11, 24]. For space limitation, we omit the technical details here.

5. Experiments

In this section, we conduct extensive experiments for image classification problems to evaluate the TSC approach.

5.1. Data Preparation

USPS, MNIST, PIE, MSRC, and VOC2007 (see Figure 1 and Table 2) are five benchmark datasets widely adopted to evaluate computer vision and patter recognition algorithms.

USPS[1] dataset consists of 7,291 training images and 2,007 test images of size 16×16.

MNIST[2] dataset has a training set of 60,000 examples and a test set of 10,000 examples of size 28×28.

From Figure 1, we see that USPS and MNIST follow very different distributions. They share 10 semantic classes, each corresponding to one digit. To speed up experiments, we construct one dataset *USPS vs MNIST* by randomly sampling 1,800 images in USPS to form the training data, and randomly sampling 2,000 images in MNIST to form the test data. We uniformly rescale all images to size 16×16, and represent each image by a 256-dimensional vector encoding the gray-scale values of all pixels. In this way, the training and test data can share the same label set and feature space.

PIE[3], which stands for "Pose, Illumination, Expression", is a benchmark face database. The database has 68 individuals with 41,368 face images of size 32×32. The face images

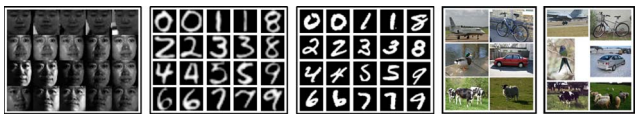


Figure 1. Examples of PIE, USPS, MNIST, MSRC, and VOC2007.

Table 2. Statistics of the six benchmark image datasets.

Dataset	Type	#Examples	#Features	#Classes
USPS	Digit	1,800	256	10
MNIST	Digit	2,000	256	10
PIE1	Face	2,856	1,024	68
PIE2	Face	3,329	1,024	68
MSRC	Photo	1,269	240	6
VOC2007	Photo	1,530	240	6

were captured by 13 synchronized cameras and 21 flashes, under varying poses, illuminations, and expressions.

In the experiments, we adopt two preprocessed versions of PIE[4],*i.e.*, **PIE1** [4] and **PIE2** [3], which are generated by randomly sampling the face images from the near-frontal poses (C27) under different lighting and illumination conditions. We construct one dataset *PIE1 vs PIE2* by selecting all 2,856 images in PIE1 to form the training data, and all 3,329 images in PIE2 to form the test data. Due to the variations in lighting and illumination, the training and test data can follow different distributions in the same feature space.

MSRC[5] dataset is provided by Microsoft Research Cambridge, which contains 4,323 images labeled by 18 classes.

VOC2007[6] dataset (the training/validation subset) contains 5,011 images annotated with 20 concepts.

From Figure 1, we see that MSRC and VOC2007 follow very different distributions, since MSRC is from standard images for evaluations, while VOC2007 is from digital photos in Flickr[7]. They share the following 6 semantic classes: "aeroplane", "bicycle", "bird", "car", "cow", "sheep". We construct one dataset *MSRC vs VOC* by selecting all 1,269 images in MSRC to form the training data, and all 1,530 images in VOC2007 to form the test data. Then following [19], we uniformly rescale all images to be 256 pixels in length, and extract 128-dimensional dense SIFT (DSIFT) features using the VLFeat open package [18]. A 240-dimensional codebook is created, where K-means clustering is used to obtain the codewords. In this way, the training and test data are constructed to share the same label set and feature space.

[1] http://www-i6.informatik.rwth-aachen.de/~keysers/usps.html

[2] http://yann.lecun.com/exdb/mnist

[3] http://vasc.ri.cmu.edu/idb/html/face

[4] http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html

[5] http://research.microsoft.com/en-us/projects/objectclassrecognition

[6] http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007

[7] http://www.flickr.com

5.2. Experimental Setup

5.2.1 Baseline Methods

We compare the TSC approach with five state-of-the-art baseline methods for image classification, as shown below.

- Logistic Regression (LR)
- Principle Component Analysis (PCA) + LR
- Sparse Coding (SC) [9] + LR
- Graph Regularized SC (GraphSC) [24] + LR
- Our proposed MMD Regularized SC (TSC_{MMD}) + LR

All SC, GraphSC, TSC_{MMD}, and TSC algorithms can learn sparse representations for input data points. In particular, SC is a special case of TSC with $\mu = \gamma = 0$, GraphSC is a special case of TSC with $\mu = 0$, TSC_{MMD} is a special case of TSC with $\gamma = 0$. Note that, our proposed TSC_{MMD} is essentially different from the method introduced in Quanz *et al*. [16, 17], which adopts kernel density estimation (KDE) to estimate the PDFs of distributions and then minimizes the Jensen-Shannon divergence between them. This is a stricter regularization than MMD and may be prone to overfitting.

5.2.2 Implementation Details

Following [24, 14], SC, GraphSC, TSC_{MMD}, and TSC are performed on both labeled and unlabeled data as an unsupervised dimensionality reduction procedure, then a supervised LR classifier is trained on labeled data to classify unlabeled data. We apply PCA to reduce the data dimensionality by keeping 98% information in the largest eigenvectors, and then perform all above algorithms in the PCA subspace.

Under our experimental setup, it is impossible to automatically tune the optimal parameters for the target classifier using cross validation, since the labeled and unlabeled data are sampled from different distributions. Therefore, we evaluate the five baseline methods on our datasets by empirically searching the parameter space for the optimal parameter settings, and report the best results of each method. For LR[8], we set the trade-off parameter C by searching $C \in \{0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100\}$. For SC[9] [9] based methods, we set the #basis vectors as $k = 128$. For GraphSC[10] [24], we set the trade-off parameter γ by searching $\gamma \in \{0.01, 0.1, 1, 10, 100\}$. For dimensionality reduction methods, we use ℓ_2-norm normalized feature vectors.

The TSC approach has three model parameters: MMD regularization parameter μ, graph regularization parameter γ, and sparsity regularization parameter λ. In the coming sections, we provide empirical analysis on parameter sensitivity, which verifies that TSC can achieve stable performance under a wide range of parameter values. When comparing with the baseline methods, we use the following parameter settings: $k = 128$, $p = 5$, $\mu = 10^5$, $\gamma = 1$, $\lambda = 0.1$, and #iterations $T = 100$. We run TSC 10 repeated times to remove any randomness caused by random initialization.

We use classification *Accuracy* on test data as the evaluation metric, which is widely used in literature [17, 24, 16]

$$Accuracy = \frac{|\mathbf{x} : \mathbf{x} \in \mathcal{D}_{ts} \wedge \widehat{y}(\mathbf{x}) = y(\mathbf{x})|}{|\mathbf{x} : \mathbf{x} \in \mathcal{D}_{ts}|}$$

where $\mathcal{D}_{ts}$ is the set of test data, $y(\mathbf{x})$ is the truth label of $\mathbf{x}$, $\widehat{y}(\mathbf{x})$ is the label predicted by the classification algorithm.

Dataset	USPS vs MNIST	PIE1 vs PIE2	MSRC vs VOC
LR	31.70±0.00	29.53±0.00	34.38±0.00
PCA	32.15±0.00	28.93±0.00	32.75±0.00
SC [9]	36.90±0.65	17.74±0.85	30.28±0.93
GraphSC [24]	41.18±0.15	19.72±1.55	30.61±0.34
TSC_{MMD}	47.30±2.13	36.71±1.76	34.27±0.45
TSC	**57.77±1.69**	**37.30±1.68**	**36.47±0.40**

Table 3. Classification accuracy (%) on cross-distribution datasets.

5.3. Experimental Results

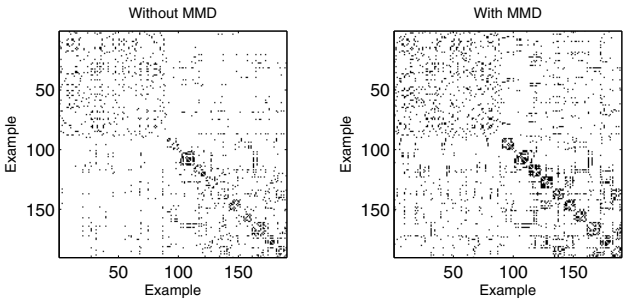
The classification accuracy of TSC and the five baseline methods on the three cross-distribution image datasets *USPS vs MNIST*, *PIE1 vs PIE2*, and *MSRC vs VOC* is illustrated in Table 3. From the results we observe that TSC achieves much better performance than the first four baseline methods. The average classification accuracies of TSC on the three datasets are 57.77%, 37.30%, and 36.47%, respectively. The performance improvements are 16.59%, 7.77%, and 2.09% compared to the best baseline methods GraphSC and LR, respectively. Furthermore, from the results averaged by 10 repeated runs in Table 3, we see that the deviations are small compared to the accuracy improvements, which validates that TSC performs stably to the random initialization. This verifies that TSC can construct robust sparse representations for classifying cross-distribution images accurately.

We have noticed that our TSC_{MMD} approach, which is a special case of TSC with $\gamma = 0$, also outperforms all the first four baseline methods. This validates that minimizing the distribution divergence is very important to make the induced representations robust for cross-distribution image classification. In particular, TSC_{MMD} has significantly outperformed GraphSC, which indicates that minimizing the distribution divergence is more important than preserving the geometric structure when labeled and unlabeled images are sampled from different distributions. It is expected that TSC can also achieve better performance than TSC_{MMD}. By incorporating the graph Laplacian term of coefficients into TSC, we aim to enrich the sparse representations with more discriminating power to benefit the classification problems.

[8] http://www.csie.ntu.edu.tw/~cjlin/liblinear

[9] http://ai.stanford.edu/~hllee/softwares/nips06-sparsecoding.htm

[10] http://www.cad.zju.edu.cn/home/dengcai/Data/SparseCoding.html



(a) weight matrix $\mathbf{W}$: $\mu = 0$. (b) weight matrix $\mathbf{W}$: $\mu = 10^5$.

Figure 2. Similarity between sparse codes of GraphSC and TSC.

Standard supervised learning methods, *i.e.*, LR, treat input data from different distributions as if they were sampled from the same distribution. In real applications, this strict assumption is usually violated, since labeled training data and unlabeled test data are usually collected in different time periods, or under different conditions. In this case, the optimal decision hyperplane trained from the labeled data cannot discriminate the unlabeled data effectively, leading to poor classification performance, as is shown in Table 3.

Standard dimensionality reduction methods, *i.e.*, PCA, generally achieve comparable performance as LR. PCA slightly outperforms LR on *USPS vs MNIST*, while slightly underperforms LR on the other datasets. A possible reason for preferring PCA is that it can extract a low-dimensional subspace, where the distribution divergence may be reduced to some extent. However, without explicitly reducing the distribution divergence, it is not an always-successful case.

Sparse coding based methods, *i.e.*, SC and GraphSC, have either strong or vulnerable datasets. SC and GraphSC significantly outperform PCA on *USPS vs MNIST*, while significantly underperform PCA on the other datasets. The reason for preferring SC and GraphSC is that the sparse representations can capture more succinct high-level semantics for image understanding. By taking into account the graph Laplacian regularizer, GraphSC can further outperform SC, which verifies that the geometric structure can indeed enrich the sparse representations with more discriminating power. However, since the labeled data and unlabeled data are sampled from different distributions as in our adopted datasets, SC and GraphSC may further enlarge the distribution divergence due to the sparse representation. Therefore, they may be less robust than PCA for cross-distribution classification. By extracting sparse representations and matching different distributions simultaneously, TSC and TSC_{MMD} can greatly enhance the robustness of sparse coding, shown in Table 3.

5.4. Effectiveness Verification

We verify the effectiveness of our TSC by inspecting the weight matrix $\mathbf{W}$. We visualize in Figure 2 the values of matrices obtained by running TSC on *USPS vs MNIST* with $\mu = 0$ and $\mu = 10^5$, and then computing $\mathbf{W}$ in Equation (2) on sparse representation $\mathbf{S}$. For clearer illustration, we sample 190 images in the dataset. Note that, the first 90 images are from the labeled training data while the last 100 images are from the unlabeled test data. Correspondingly, in weight matrix $\mathbf{W}$, the top-left and bottom-right blocks indicate the *within-distribution* similarity, and the top-right and bottom-left blocks indicate the *between-distribution* similarity.

Figure 2(a) shows the weight matrix $\mathbf{W}$ with $\mu = 0$. We observe that the between-distribution similarity is sparse, indicating that the distribution difference is still very large, even when feature extraction is performed. Most existing sparse coding methods, such as SC and GraphSC, have not explicitly minimized the distribution difference, resulting in unsatisfactory performance for cross-distribution problems.

Figure 2(b) shows the weight matrix $\mathbf{W}$ with $\mu = 10^5$. This time, we observe that the between-distribution similarity is greatly enriched. This naturally leads to better generalization capability, that is, with sparse representation $\mathbf{S}$, a supervised classifier trained on the labeled training data is expected to perform much better on the unlabeled test data.

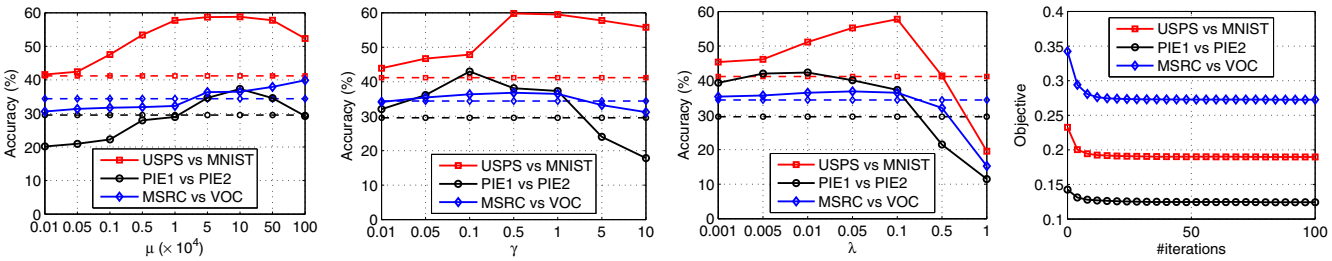
5.5. Parameter Sensitivity

We conduct empirical analysis on parameter sensitivity using all datasets, which validates that TSC can achieve optimal performance under a wide range of parameter values.

We run TSC with varying values of MMD regularization parameter μ. Theoretically, μ controls the weight of MMD regularization, and larger values of μ will make the distribution matching more important in TSC. An extreme case is $\mu \to \infty$, where only distribution matching is guaranteed, but both sparse coding and geometric preservation for the input images are discarded. Another extreme case is $\mu \to 0$, where only sparse coding and geometric preservation for input images are guaranteed, but the distribution matching is discarded. In both extreme cases, TSC cannot extract robust sparse representations for cross-distribution image classification. We plot the classification accuracy w.r.t. different values of μ in Figure 3(a), and can choose $\mu \in [10^4, 10^6]$.

We run TSC with varying values of graph regularization parameter γ. Theoretically, γ controls the weight of graph regularization, and larger values of γ will make the geometric preservation more important in TSC. An extreme case is $\gamma \to \infty$, where only the geometric preservation is guaranteed. Then TSC will degenerate to standard spectral clustering, which cannot reduce the distribution divergence. Another extreme case is $\gamma \to 0$, where the geometric preservation is discarded. Then TSC will degenerate to TSC_{MMD}, which cannot enrich the new representations with discriminating power. We plot the classification accuracy w.r.t. different values of γ in Figure 3(b), and choose $\gamma \in [0.01, 1]$.

We run TSC with varying values of sparsity regulariza-



(a) MMD regularization μ (b) graph regularization γ (c) sparsity regularization λ (d) objective w.r.t. #iterations

Figure 3. Parameter sensitivity analysis and convergence study of the proposed TSC approach (dashed lines show the best baseline results).

tion parameter λ. Theoretically, λ controls the complexity of coding matrix $\mathbf{S}$, and can prevent TSC from over-fitting the input data or degenerating to trivial solutions during the iterative procedure. More importantly, λ also controls the sparsity level of TSC. When $\lambda \to 0$, TSC will be ill-posed since the over-complete approximation ($k \geq m$) cannot be well defined without proper regularization. On the contrary, when $\lambda \to \infty$, TSC will be dominated by the sparsity regularization and the important properties of input data are not captured. We plot the classification accuracy w.r.t. different values of λ in Figure 3(c), and can choose $\lambda \in [0.001, 0.1]$.

5.6. Convergence Study

Since TSC is an iterative algorithm, we empirically check its convergence property. Figure 3(d) shows that the objective value (averaged by #examples) decreases steadily with more iterations and converges within 100 iterations.

6. Conclusion

In this paper, we propose a novel Transfer Sparse Coding (TSC) approach for robust image representation. An important advantage of TSC is the robustness to the distribution difference between the labeled and unlabeled images, which can substantially improve cross-distribution image classification problems. Extensive experimental results on several benchmark datasets show that TSC can achieve superior performance against state-of-the-art sparse coding methods.

References

[1] M. Aharon, M. Elad, A. Bruckstein, and Y. Katz. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11), 2006. 2, 4

[2] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems 15*, NIPS, 2001. 2

[3] D. Cai, X. He, and J. Han. Spectral regression: A unified approach for sparse subspace learning. In *Proceedings of the 7th IEEE International Conference on Data Mining*, ICDM, 2007. 5

[4] D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011. 5

[5] R. Fletcher. *Practical methods of optimization*. Wiley-Interscience, 1987. 4

[6] S. Gao, I. W.-H. Tsang, L.-T. Chia, and P. Zhao. Local features are not lonely – laplacian sparse coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2010. 1, 2, 3

[7] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Scholkopf, and A. J. Smola. A kernel method for the two-sample problem. In *Advances in Neural Information Processing Systems 20*, NIPS, 2006. 2, 3

[8] K. Huang and S. Aviyente. Sparse representation for signal classification. In *Advances in Neural Information Processing Systems 21*, NIPS, 2007. 1

[9] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems 20*, NIPS, 2006. 1, 2, 3, 4, 5, 6

[10] Y. N. Liu, F. Wu, Z. H. Zhang, Y. T. Zhuang, and S. C. Yan. Sparse representation using nonnegative curds and whey. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2010. 1, 2

[11] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the International Conference on Machine Learning*, ICML, 2009. 1, 2, 3, 4, 5

[12] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *Advances in Neural Information Processing Systems 23*, NIPS, 2009. 1

[13] S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, AAAI, 2008. 1, 3

[14] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011. 1, 3, 6

[15] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010. 1, 2

[16] B. Quanz, J. Huan, and M. Mishra. Knowledge transfer with low-quality data: A feature extraction issue. In *Proceedings of the IEEE International Conference on Data Engineering*, ICDE, 2011. 1, 2, 6

[17] B. Quanz, J. Huan, and M. Mishra. Knowledge transfer with low-quality data: A feature extraction issue. *IEEE Transactions on Knowledge and Data Engineering*, 24(10), 2012. 1, 2, 6

[18] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008. 5

[19] C. Wang, D. Blei, and L. Fei-Fei. Simultaneous image classification and annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2009. 5

[20] J. J. Wang, J. C. Yang, K. Yu, F. J. Lv, T. Huang, and Y. H. Gong. Locality-constrained linear coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2010. 1, 2

[21] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2), 2009. 1

[22] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2009. 1

[23] M. Yang, L. Zhang, J. Yang, and D. Zhang. Robust sparse coding for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2011. 1

[24] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai. Graph regularized sparse coding for image representation. *IEEE Transactions on Image Processing*, 20(5), 2011. 1, 2, 3, 4, 5, 6