3-2013

# Robust image analysis with sparse representation on quantized visual features

Bingkun BAO

Guangyu ZHU

Jialie SHEN

*Singapore Management University*, jlshen@smu.edu.sg

Shuicheng YAN

**DOI:** https://doi.org/10.1109/TIP.2012.2219543

Published in IEEE Transactions on Image Processing, 2013 March, Volume 22, Issue 3, Pages 860-871
https://doi.org/10.1109/TIP.2012.2219543

# Robust Image Analysis With Sparse Representation on Quantized Visual Features

Bing-Kun Bao, Guangyu Zhu, Jialie Shen, and Shuicheng Yan, *Senior Member, IEEE*

***Abstract*— Recent techniques based on sparse representation (SR) have demonstrated promising performance in high-level visual recognition, exemplified by the highly accurate face recognition under occlusion and other sparse corruptions. Most research in this area has focused on classification algorithms using raw image pixels, and very few have been proposed to utilize the quantized visual features, such as the popular bag-of-words feature abstraction. In such cases, besides the inherent quantization errors, ambiguity associated with visual word assignment and misdetection of feature points, due to factors such as visual occlusions and noises, constitutes the major cause of dense corruptions of the quantized representation. The dense corruptions can jeopardize the decision process by distorting the patterns of the sparse reconstruction coefficients. In this paper, we aim to eliminate the corruptions and achieve robust image analysis with SR. Toward this goal, we introduce two transfer processes (ambiguity transfer and mis-detection transfer) to account for the two major sources of corruption as discussed. By reasonably assuming the rarity of the two kinds of distortion processes, we augment the original SR-based reconstruction objective with $\ell_0$-norm regularization on the transfer terms to encourage sparsity and, hence, discourage dense distortion/transfer. Computationally, we relax the nonconvex $\ell_0$-norm optimization into a convex $\ell_1$-norm optimization problem, and employ the accelerated proximal gradient method to optimize the convergence provable updating procedure. Extensive experiments on four benchmark datasets, Caltech-101, Caltech-256, Corel-5k, and CMU pose, illumination, and expression, manifest the necessity of removing the quantization corruptions and the various advantages of the proposed framework.**

***Index Terms*— Image classification, quantized visual feature, sparse representation.**

Manuscript received February 22, 2012; revised June 6, 2012; accepted September 4, 2012. Date of publication September 21, 2012; date of current version January 21, 2013. This work was supported in part by the National Program on Key Basic Research Project 973 Program, under Project 2012CB316304, the National Natural Science Foundation of China under Grant 61201374, the China Postdoctoral Science Foundation under Grant 2011M500430, the Singapore Ministry of Education through the K. C. Wong Education Foundation under Grant MOE2010-T2-1-087. The work of J. Shen was supported in part by the Mobile plus Cloud Computing Theme Research Program Award - Microsoft Research, 2010–2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Wan-Chi Siu.

B.-K. Bao is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the China-Singapore Institute of Digital Media, 119613 Singapore (e-mail: bingkunbao@gmail.com).

G. Zhu is with the University of California at Berkeley, Berkeley, CA 94720 USA (e-mail: guangyu_zhu@mfe.berkeley.edu).

J. Shen is with Singapore Management University, 188065 Singapore (e-mail: jlshen@smu.edu.sg).

S. Yan is with the National University of Singapore, 117576 Singapore (e-mail: eleyans@nus.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2012.2219543

## I. Introduction

THE TECHNIQUE of Sparse Representation (SR)[1] has lent itself to numerous applications in image processing and image analysis recently [1], *e.g.*, image restoration [2], image super-resolution [3], visual recognition [4], [5], and image annotation [6]. The principal idea underpinning these applications is embarrassingly simple: any input sample is approximated in terms of a sparse linear combination of atomic samples collected in a given overcomplete dictionary. These combination coefficients and practically also the identified noises are used for the subsequent visual processing and reasoning. Computationally, pursuit for such sparsest representations boils down to linear programming (LP, for data without noises) or quadratic constrained linear programming (QCLP, for data contaminated with noises) problems on $\ell_1$-norm minimization, as a surrogate to the desired $\ell_0$-norm minimization [7]. Recent research has shown that SR appears to be biologically plausible as well as empirically effective [8], [9].

Of particular interest to the current paper is the application of SR into visual recognition. In this regard, a frequently cited example is the robust high-accuracy face recognition algorithm presented by Wright *et al.* [4]. The input to that algorithm is the vectorized raw image pixels, and experiments therein have demonstrated in face recognition even this low-level simplistic raw feature can guarantee satisfactory recognition performance under mild conditions, powered by SR. Nevertheless, this may not be the case for many other recognition tasks. For example, state-of-the-art object recognition algorithms normally involve extraction and encoding of image structures and regions of interest, and many discriminative classification models generate a holistic feature representation by feature quantization and summarization (see, e.g., chapter 14 in [10]). Examples abound in the vein. Low-level representation often involves the various kinds of salient points (corners, blobs, edges, etc.) and their descriptors such as the SIFT descriptor [11], Local Binary Patterns (LBPs) [12], Histogram of Oriented Gradients (HOGs) [13], where the quantization and encoding are normally with respect to some pre-defined magnitudes, orientations, or mixture of them. Mid-level representation has seen the popularity of the Bag-of-Words (BoW) method [13], in which the quantization centroids are most often learnt from data. Despite the diversity and hierarchy, a common complication

[1]In signal processing community and others, SR is also widely known as compressive sensing, compressed sensing, or sparse coding with weak distinctions. We will use these terms interchangeably.

of these types of visual feature encoding, as compared to the raw feature representation, is the vector quantization process. Quantization conceivably entails inherent information loss, and moreover sensitivity to noise and outliers, causing dense errors in representation even with sparse noises and outliers. These factors make SR-based robust classification with quantized visual features more tricky and challenging, and the current work is aimed at providing a timely investigation and removing the obstacle. To make the investigation concrete and elucidating, we focus on discussing about the mid-level BoW representation as the first step, and expect extensions to other quantization situations to be straightforward.

### A. Problems With Quantized Visual Features

The process of BoW encoding towards representation is essentially vector quantization. During this process, each raw visual feature vector is assigned to its nearest prototype centroid, and the count for the corresponding centroid is increased by unity. A final normalization is normally taken on the counting vector to form a normalized (e.g., $\ell_1$-norm to be unit) histogram feature representation. Obviously there is inevitable information loss due to the assignment, commonly known as the quantization error. While this kind of error can be partially reduced by increasing the number of quantization centroids/bins, we are mainly interested in the orthogonal realm as depicted in the following two processes.

1) *Ambiguity Transfer*. This is the process where one raw feature vector is assigned to the wrong centroid, perhaps due to feature noise or numerical difficulty (middle-way between two or more centroids). Eventually this process will decrease the value of the true centroid bin and increase that of the falsely assigned. Notice that the probabilities of one particular raw feature vector to be assigned to unintended centroid bins are different.
2) *Mis-Detection Transfer*. This is the process where either positive raw feature structures are not detected (false negative), or negative raw feature structures are detected (false positive), possibly because of noise, occlusions or improper setting of detection parameters. Correspondingly there are missing raw feature vectors or spurious ones. The former case causes increase in value of its corresponding histogram bin and the latter causes decrease. These are further complicated by the final normalization, since sparse transfers shall result in dense errors in final histogram.

The above two transfer processes could cause serious corruptions to the quantized visual feature representation, and hence hurt the subsequent SR-based recognition. They deserve strategic treatment as we shall propose below. Before delving in, we present in Fig. 1 an illustration on causes to ambiguity and mis-detection processes, and their effect on the histogram values.

### B. Our Remedy Scheme and Contributions

We propose an $\ell_1$-norm minimization based framework to eliminate the corruptions and to achieve robust visual recognition. We first model the two transfer processes separately via proper transfer matrices[2]. By noticing the general low portion of mis-assignment and raw feature mis-detection, we adopt sparsity priors to both transfer processes, and instantiate the priors with $\ell_0$-norm regularization that encourages sparsity. This regularization augmented to the original SR recognition scheme constitutes a joint $\ell_0$-norm objective subject to a linear constraint, which can be relaxed into a convex program in the form of constrained $\ell_1$-norm minimization.

We will note that the resulting $\ell_1$-norm optimization problem takes a similar form to the popular Lasso problem in statistical learning [14], if only one of the variables is considered once. Despite the availability of dozens of efficient solution schemes for Lasso (see [15] for a brief review of state-of-the-art algorithms), we will see that these algorithms scale up poorly for the current problem. Instead, we will describe a first-order optimization scheme, based on the accelerated proximal gradient (APG) method [16], [17], to solve this particular problem efficiently. Empirically the computational cost of this customized scheme scales up gracefully with the scale of the data.

The rest of this paper is organized as follows. Section II provides the details on the causes and remedy to two kinds of corruptions and the unified formulation. The iterative optimization procedure is proposed in Section III. Section IV gives a brief discussion on the related methods. Section V presents the experimental results, and the conclusion remarks are given in Section VI. Some of the technical derivation for results to the optimization part will be deferred to the appendix.

## II. Modeling and Remedy to Corruptions in Quantized Visual Features

In this section, we describe in detail two models to describe the ambiguity transfer and mis-detection transfer processes, respectively. By imposing the sparsity priors, we arrive at two separate programs to rectify the two processes. Finally we combine these two programs with the SR-based reconstruction and obtain a unified formulation that is expected to achieve simultaneous corruption removal and robust analysis. Before delving in, we fix our notation as follows. We use normal lowercase and capital letters for vectors and matrices, respectively, e.g., $s \in \mathbb{R}^n$ and $K \in \mathbb{R}^{m\times n}$. In particular, we define $\mathbb{1}$ to be a vector of all one's, whose dimension will be clear from context. For vector and matrix norms, we will be particularly interested in the $\ell_0$-norm[3] $\|\cdot\|_0$ which counts the number of non-zero elements in a vector or matrix and the $\ell_1$-norm $\|\cdot\|_1$ which is the summation of absolute values of all elements in a vector or matrix. The Frobenius norm $\|\cdot\|_F$ for matrices is a straightforward extension of the $\ell_2$-norm for vectors. Matrix inner product w.r.t. the Frobenius norm will appear as conventional, i.e., $\langle A, B\rangle = \mathrm{tr}\left(A^\top B\right)$, where $\mathrm{tr}\,(\cdot)$ denotes the trace of a square matrix. Other rare notations will be defined from context.

[2]These two processes can be decoupled and analyzed separately.

[3]In fact it is not a valid norm in the algebraic sense.

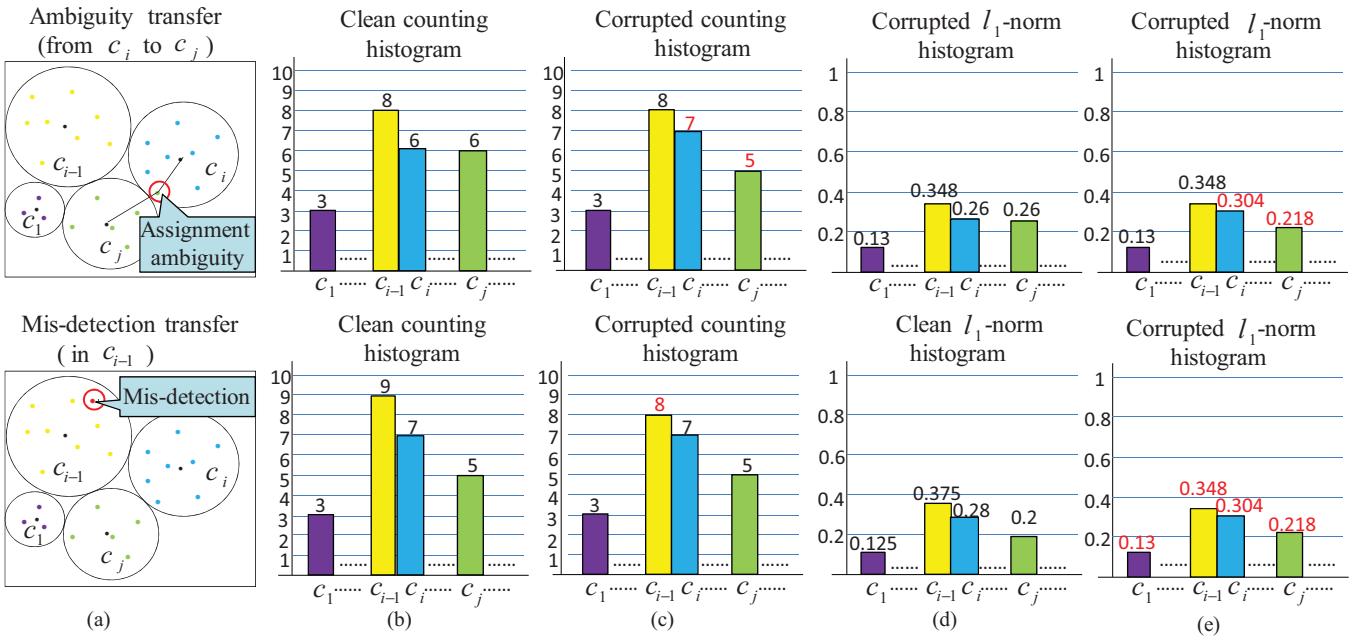


Fig. 1. Illustration on the causes and effects of ambiguity transfer and mid-detection transfer processes. (a) Assignment ambiguity causes the ambiguity transfer; features that are spurious or missing from detection cause the mis-detection transfer. (b) Example clean counting histogram of features. (c) Counting histogram effected using ambiguity transfer (top) and misdetection transfer (bottom). (d) and (e) Clean $\ell_1$ normalized histogram and the effected ones, corresponding to (b) and (c), respectively.

### A. Model of Ambiguity Transfer and Mis-Detection Transfer

The ambiguity transfer arises from the difficulty to assign a raw feature vector to the correct centroid, and hence causes shift of the histogram values from one bin to another. To quantitatively model this shift process, we introduce an "ambiguity transfer matrix" $K$. The element $K_{ij}$ indicates the amount of shift in normalized histogram values from the $j^{th}$ to the $i^{th}$ bin[4]. Therefore, $(K\mathbb{1})_i$ amounts to the total increase of value to the $i^{th}$ bin, due to the shift of values from all the other bins; similarly $(K^\top\mathbb{1})_i$ represents the amount shifted from the $i^{th}$ bin to all the others. Obviously, the resultant change due to the ambiguity transfer on the $i^{th}$ histogram bin is $(K\mathbb{1})_i-(K^\top\mathbb{1})_i$. Thus, the overall corruptions to the histogram vector $\mathcal{E}_A$ is formulated as

$$\mathcal{E}_A = K\mathbb{1} - K^\top\mathbb{1}. \tag{1}$$

*Formulation 2.1 (Model of Ambiguity Transfer):* Assume assignment ambiguity only corrupts a relatively small portion of histogram bins. The ambiguity transfer of the whole histogram is modeled as

$$\mathcal{E}_A = K\mathbb{1} - K^\top\mathbb{1}. \tag{2}$$

Spurious presence and missing of raw features make another source of histogram corruptions, which we collectively term as the mis-detection transfer. Causes to mis-detection can be diverse and complicated. For example, during feature extraction in image analysis, prevalent image noise can easily fail the feature detector; even the image itself is clean, feature detectors are normally controlled by several parameters, such as the scale, orientation, explicit thresholding, etc. In actual practice, adaptability of feature detectors is far less rich and versatile than the image generating process, directly leading to unsatisfactory detection.

Mis-detection quantization can be separated into two steps: 1) corrupting a certain histogram-bin, and 2) propagating the corruption by normalization. For a certain image, the corrupted histogram vector is denoted as $y = [y_1, y_2, \ldots, y_r]^T$. We assume the corresponding non-corrupted histogram vector is $x = [x_1, x_2, \ldots, x_r]^T$. To model the first process, we introduce the "mis-detection vector" $p = [p_1, \ldots, p_r]^T$, where the element $p_i$ indicates the unnormalized amount of corruption in $i$-th histogram-bin. Then non-corrupted histogram vector $x = [x_1, \ldots, x_r]^T$ is changed to $[x_1 + p_1, \ldots, x_r + p_r]^T$. In the second step, $x$ is finally normalized to $[\frac{x_1+p_1}{1+\sum_{i=1}^r p_i}, \ldots, \frac{x_r+p_r}{1+\sum_{i=1}^r p_i}] = y$. So, $x_i = (1+\sum_{i=1}^r p_i)y_i - p_i$, or collectively $x = (1 + \sum p_i)y - p$. The overall corruptions $\mathcal{E}_M$ in $y$ is formulated as

$$\mathcal{E}_M = y - x = -\sum_{i=1}^{r} p_i y + p = (\mathbb{I} - y\mathbb{1}^\top)p \tag{3}$$

where $\mathbb{I}$ denotes the identity matrix of proper dimension. Let $B = \mathbb{I} - y\mathbb{1}^\top$, referred as a "mis-detection transfer matrix". Thus, mis-detection transfer can be formulated as following.

*Formulation 2.2 (Model of Mis-Detection):* Assume mis-detection only corrupts a small portion of histogram bins. Then the mis-detection transfer the whole histogram is modeled as

$$\mathcal{E}_M = Bp, \qquad \text{where } B = \mathbb{I} - y\mathbb{1}^\top. \tag{4}$$

### B. Remedy to Ambiguity Transfer and Mis-Detection Transfer

Under the known of ambiguity transfer value $\mathcal{E}_A$, we do not expect too much ambiguity associated with vector assignment (otherwise it means we have generated a bad

[4]Henceforth, we will assume these bins are ordered according to ordering of their corresponding centroids.

set of quantization centroids and need to refine the choice). This implies that there are only a relatively small portion of histogram bins affected by the ambiguity shift. Hence we can reasonably adopt the sparsity prior to the transfer matrix $K$. Mathematically we hope the number of nonzero elements in $K$, or exactly the $\|K\|_0$ to be small. On the other hand, we have to take care of the locality nature of mis-assignment. In other words, assignment ambiguity normally occurs when the query feature vector has similar distances to two or more centroids. This mostly happens when these centroids are close to each other, whence minor noise or corruption to the query vector will simply cause wrong assignment. By contrast, this kind of ambiguity is rare for centroids that are far apart. The modeled ambiguity transfer should be with the ability to correct the error from nearby centroid and avoid the error shift from distant centroid, therefore, we give a larger penalty for distant centroids to suppress this kind of ambiguity transfer in our model. In accordance with this, we introduce a weighting matrix $W$ to weight transfer amount in $K$ delicately based on their mutual distances. Specifically, given the $i^{th}$ centroid $c_i$ and the $j^{th}$ centroid $c_j$, and a bandwidth parameter $\sigma$, we define the corresponding weight to be

$$W_{ij} = 1 - \exp\left(-\frac{\|c_i - c_j\|^2}{\sigma^2}\right) \tag{5}$$

which is large for distant centroids and small for nearby centroids. Hence continuing with the argument for instantiation of the sparsity prior, we now want the weighted transfer counts $\|W \odot K\|_0$ to be small, where $\odot$ denotes the element-wise matrix multiplication. Note that the non-vanishing values of $K$ tends to concentrate on elements modeling nearby centroids (upon correct identification of the corrupted bins by assignment ambiguity), whereas $W$ will put a small weight on these elements. Hence towards minimization of the objective, we expect assignment ambiguity to be rare, concurring with our previous assumption. Thus, we can formulate the following objective function to remedy to ambiguity transfer

$$\begin{aligned} \min_{K} \quad & \|W \odot K\|_0. \\ s.t. \quad & \mathcal{E}_A = K\mathbb{1} - K^\top \mathbb{1}. \end{aligned} \tag{6}$$

However, direct search for the sparsest solution leads to combinatorial problems that are **NP** hard [18]. Nevertheless, recent development of compressed sensing reveals that $\ell_1$-norm minimization can be an effective convex surrogate to $\ell_0$-norm minimization, when the desired solution is sparse enough [19]. Under mind conditions, the surrogate can find out the exact solution to the $\ell_0$-norm minimization problem. Hence instead of minimizing $\|W \odot K\|_0$ directly, we seek to

$$\begin{aligned} \min_{K} \quad & \|W \odot K\|_1 \\ \text{s.t.} \quad & \mathcal{E}_A = K\mathbb{1} - K^\top \mathbb{1}. \end{aligned} \tag{7}$$

With the known of mis-detection value $\mathcal{E}_M$, if we reasonably assume cases of feature mis-detection are rare and only a small proportion of histogram bins are affected by this transfer, we expect $p$ to be sparse in practice. Running the similar argument about the surrogation of the $\ell_1$-norm to the $\ell_0$-norm for sparse recovery, we arrive at the following formulation with the corruption removal objective

$$\begin{aligned} \min_{p} \quad & \|p\|_1 \\ \text{s.t.} \quad & \mathcal{E}_M = Bp. \end{aligned} \tag{8}$$

### C. Unified Objective Optimization Formulation for RIASR

We set out to translate the SR-based robust recognition framework on raw image features [4] to that on quantized visual representation, mostly mid-level histograms. To accomplish this, we base our computational model on the original SR framework, meanwhile handling the distinctive forms of corruptions, i.e., ambiguity transfer and mis-detection transfer, with innovative ingredients. We will unify the SR reconstruction framework and our corruption removal methods as described in Formulations 2.1 and 2.2 in this section.

For a typical classification problem, let matrix $D$ collect all the training samples, i.e., $D = [x_1, \ldots, x_n]$ for $x_i \in \mathbb{R}^r_+, \forall i \in [1, \ldots, n]$, where $\mathbb{R}_+$ indicates non-negative real number. Here $x_i$ is a quantized histogram representation for each sample. For a given test sample $y$, we expect a sparse linear reconstruction over the training samples in the form $y = Ds + \mathcal{E}$, where $s$ is the reconstruction coefficient vector that is expected to be sparse, and $\mathcal{E}$ accounts for errors and corruptions associated with $y$. Appendix VI-A shows that the corruptions caused by ambiguity transfer and that caused by mis-detection transfer are linearly additive, hence we can decompose the error term further as $\mathcal{E} = \mathcal{E}_A + \mathcal{E}_M$, that is

$$y = Ds + \mathcal{E}_A + \mathcal{E}_M. \tag{9}$$

Substituting the analytic forms of these error terms as proposed in Formulation 2.1 and 2.2 and collecting all the sparsity-encouraging objectives, we arrive at a unified optimization problem for all our purposes:

$$\begin{aligned} \min_{s,K,p} \quad & \|s\|_1 + \lambda_1 \|K \odot W\|_1 + \lambda_2 \|p\|_1 \\ \text{s.t.} \quad & y = Ds + \left[K\mathbb{1} - K^\top \mathbb{1} + Bp\right], \qquad s \geq 0. \end{aligned} \tag{10}$$

For the sake of treatment, we make a trivial change of variables by making $E = K \odot W$, then the program we will attack in the rest of the paper will be

$$\begin{aligned} \min_{s,E,p} \quad & \|s\|_1 + \lambda_1 \|E\|_1 + \lambda_2 \|p\|_1, \\ \text{s. t.} \quad & y = Ds + \left[(E./W)\mathbb{1} - (E./W)^\top \mathbb{1} + Bp\right], \quad s \geq 0. \end{aligned} \tag{11}$$

where we employ ./ to denote element-wise division of matrices henceforth and $B = \mathbb{I} - y\mathbb{1}^\top$.

## III. Practical Optimization via the Accelerated Proximal Gradient (APG) Method

In this section, we purpose a computationally efficient attack of the optimization in Eqn. (11). We start with a naive solution to the problem which can be built on the Lasso [15] and its numerous solvers, and explain about the computational limitation and then alter to our proposal. We propose an accelerate proximal gradient algorithm [16], [17] to efficiently

solve the problem. In comparison to the Lasso solution with poor scaling-up capacity, our proposal is especially promising in this aspect. Some technical deductions of results presented in this section are provided in Appendix.

### A. Two Solution Schemes

Since it is hard to handle the equality constraint about reconstruction in Eqn. (11) directly, we propose to progressively solve a subproblem in this form

$$\min_{s,E,p} \quad \|Ds - y + [(E./W)\mathbb{1} - (E./W)^\top \mathbb{1} + Bp]\|_F^2 + \mu(\|s\|_1 + \lambda_1\|E\|_1 + \lambda_2\|p\|_1), \quad \text{with} \quad s \geq 0 \tag{12}$$

each time, while gradually decreasing the value of parameter $\mu$ over iterations. This is justified in view that as $\mu \to 0$ the variant problem is approaching the original.

For Eqn. (12), it is immediately apparent that the objective can be separated into three subproblems while fixing any two unknown terms and optimizing the remaining one. While it is obvious optimizing w.r.t. $s$ and $p$ respectively boils down to standard Lasso problems (i.e., $\ell_1$ constrained linear regression, see [20]), it also holds for optimizing w.r.t. $E$ as shown below.

If one converts matrix $E = [e_1, \ldots, e_r] \in \mathbb{R}^{r\times r}$ into its vector form $e = [e_1; \ldots; e_r] \in \mathbb{R}^{r^2\times 1}$ by stacking all columns, the optimization turns out to be

$$\min_{s,e,p} \quad \|Ds - y + Ae + Bp\|_F^2 + \mu(\|s\|_1 + \lambda_1\|e\|_1 + \lambda_2\|p\|_1), \quad \text{with} \quad s \geq 0, \tag{13}$$

where $B = \mathbb{I} - y\mathbb{1}^\top$, and $A \in \mathbb{R}^{r\times r^2}$ is defined as

$$A_{i,j} = \begin{cases} \frac{1}{W_{i,j}} & \text{if} \quad j = 1, 2, \cdots, r \,\&\, j \neq i, \\ -\frac{1}{W_{i,\frac{j-i}{r}+1}} & \text{if} \quad j = i, i+r, \cdots, i+(r-1)r, \\ 0 & \text{otherwise.} \end{cases}$$

Hence if one wishes to optimize $e$ while fixing $s$ and $p$, one can apply the "flatting" trick as introduced above and convert back and forth between the vector and matrix forms. This however immediately reveals one potential problem: the dimension of the vector will grows very quickly with the dimension of the matrix, as dictated by the $r^2$ term.

We wish to avoid the awkward conversion that causes the dimensionality problem, and also to employ simple optimization techniques with good convergence rate guarantee. The shrinkage operator for matrices recently employed in several works, e.g., [21], and the innovative first-order method introduced by Nestorov [22], [23] combined with the proximal gradient method jointly offer the promise. In fact, the accelerate proximal gradient (APG) method has been described and applied with success in several remarkable pieces of reports, e.g., [16], [17], and [24]. We will next briefly review the basics of APG method and several useful shrinkage operator results, before presenting our solution scheme that exhibits excellent scalability and convergence. Meanwhile we would like to note that the first-order acceleration method has recently been employed to solve the Lasso problem with insightful analysis [15]. There instead of applying the proximal method to the smooth term only, the authors employ the smoothing technique as presented in Nestorov [23] and smoothed the $\ell_1$ norm which is not smooth.

**Algorithm 1** Accelerate Proximal Gradient Method

1: **While** *not converged* **do**
2: $Y_k \leftarrow X_k + \frac{b_{k-1}-1}{b_k}(X_k - X_{k-1})$;
3: $G_k \leftarrow Y_k - \frac{1}{L_f}\nabla f(Y_k)$;
4: $X_{k+1} \leftarrow \arg\min_X \left\{\mu g(X) + \frac{L_f}{2}\|X - G_k\|^2\right\}$;
5: $b_{k+1} \leftarrow \frac{1+\sqrt{4b_k^2+1}}{2}$, $k \leftarrow k+1$;
6: **end**

### B. The Accelerate Proximal Gradient Method and Shrinkage Operators

Given an unconstrained convex problem

$$\min_{X\in\mathcal{H}} F(X) \doteq \mu g(X) + f(X) \tag{14}$$

for a real Hilbert space $\mathcal{H}$ endowed with an inner product $\langle\cdot,\cdot\rangle$ and a corresponding norm $\|\cdot\|$ and dual norm $\|\cdot\|^*$, with $\mu > 0$ being a balancing parameter. Suppose both $g(X)$ and $f(X)$ are convex, and further $f(X)$ is continuously differentiable and $\nabla f(X)$ is Lipschitz continuous with constant $L_f$, i.e.,

$$\|\nabla f(X_1) - \nabla f(X_2)\|^* \leq L_f\|X_1 - X_2\|. \tag{15}$$

Instead of directly minimizing $F(X)$, proximal gradient algorithms minimize a sequence of separable local quadratic approximations to $F(X)$ (specifically local approximation to the smooth term $f(X)$ while keeping $g(X)$ intacted), denoted as $Q(X, Y)$ which is formed at tactically chosen points $Y$:

$$Q(X,Y) \doteq f(Y) + \langle\nabla f(Y), X - Y\rangle + \frac{L_f}{2}\|X - Y\|^2 + \mu g(X). \tag{16}$$

Let $G = Y - \frac{1}{L_f}\nabla f(Y)$, then

$$X = \arg\min_X Q(X,Y) = \arg\min_X \left\{\mu g(X) + \frac{L_f}{2}\|X - G\|^2\right\}. \tag{17}$$

Hence to solve the optimization in (14), one may repeatedly set $X_{k+1} = \arg\min_X Q(X, Y_k)$ with $Y_k$ chosen based on the sequence $X_0, \ldots, X_k$, and the convergence of this iterative process depends strongly on the choice of points $Y_k$ on which local approximations $Q(X, Y)$ are formed. The computational complexity is $O(L/\epsilon)$ [16]. The general accelerated version of the proximal gradient method is presented in Algorithm 1.

The major purposes of forming sequentially the separable quadratic approximations in proximal methods lie at that many cases of interest admit simple or even closed-form solutions to the proximal optimization in Eqn (17). And this dictates the per-iteration complexity for the whole problem whatever convergence rate obtained. Hence before applying APG to our particular problem, we present two closed-form solutions to two generic optimization problems. First we introduce two shrinkage (also termed as soft-thresholding operators in some

articles [25]) operators. Specifically, for matrix $X \in \mathbb{R}^{m\times n}$ and $\epsilon > 0$, we have the symmetric shrinkage operator

$$S_\epsilon[X] = \begin{cases} X_{ij} - \epsilon & X_{ij} > \epsilon, \\ X_{ij} + \epsilon & X_{ij} < -\epsilon, \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

and the asymmetric shrinkage operator

$$T_\epsilon[X] = \begin{cases} X_{ij} - \epsilon & X_{ij} > \epsilon, \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

where we use the subscript $ij$ to denotes element-wise operations to all elements of the matrix $X$. And simple exercising of the classic shrinkage operator analysis (e.g., [26]) reveals that the above shrinkage operators are closed-form solutions to two optimization problems of particular interest to our work.

*Proposition 3.1 [20]:* For matrix $X \in \mathbb{R}^{m\times n}$, we have

$$S_\epsilon[G_k] = \arg\min_X \epsilon\|X\|_1 + \frac{1}{2}\|X - G_k\|_F^2. \tag{20}$$

$$T_\epsilon[G_k] = \arg\min_{X\geq 0} \epsilon\|X\|_1 + \frac{1}{2}\|X - G_k\|_F^2. \tag{21}$$

One can easily verify these results by noticing that the optimizations are separable w.r.t. matrix elements and hence the problems each reduce to scalar optimization ones.

### C. Accelerate Proximal Gradient Method for RIASR

We are now ready to zoom in to our particular problem and apply the APG algorithm. By comparing Eqn. (12) and Eqn. (14), we have the component objectives as

$$\begin{aligned} f(X) &= \left\| Ds - y + [(E./W)\mathbb{1} - (E./W)^\top\mathbb{1} + Bp] \right\|_F^2 \\ g(X) &= \|s\|_1 + \lambda_1\|E\|_1 + \lambda_2\|p\|_1 \end{aligned} \tag{22}$$

where $X = (s, E, p)^\top$ and $B = \mathbb{I} - y\mathbb{1}^\top$ as a constant matrix. We note that we still have one additional constraint $s \geq 0$, but we can efficiently handle them in the respective proximal problems (as shown in detail in the appendix). Formally, by composition rules of convexity (see e.g., Boyd and Vandenberghe [27]), we assert the convexity of both $f(X)$ and $g(X)$ and further the smoothness of $f(X)$.

Next we will show that $f(X)$ in Eqn. (22) satisfies Lipschitz continuous condition and also identify the Lipschitz constant.

*Proposition 3.2:* $f(X)$ defined in Eqn. (22) is Lipschitz continuous with constant

$$\begin{aligned} L_f = \max\{&\sqrt{16\|D\|^4 + 64\|1./W\|^2\|D\|^2 + 16\|B\|^2\|D\|^2}, \\ &\sqrt{32m\|D\|^2\|1./W\|^2 + 128m^2\|1./W\|^4 + 32m\|B\|^2\|1./W\|^2}, \\ &\sqrt{16\|D\|^2\|B\|^2 + 64m^2\|B\|^2\|1./W\|^2 + 16\|B\|^4}\} \end{aligned} \tag{23}$$

where we have for the moment reserved $\|\cdot\|$ to denote the operator norm (i.e., the largest singular value) of a matrix.

*Proof:* Noting that $\|A\|_F^2 = \mathrm{tr}\left(A^\top A\right)$ and $\mathrm{tr}\,(AB) = \mathrm{tr}\,(BA)$ and some properties in trace derivatives, we can calculate $\nabla f(X)$ based on Eqn. (22) as

$$\begin{aligned} \nabla_s f(X) &= 2D^\top Ds + 2D^\top\Big[-y + (E./W)\mathbb{1} - (E./W)^\top\mathbb{1} + Bp\Big]; \\ \nabla_E f(X) &= 2\Big[(E./W)\mathbb{1} - (E./W)^\top\mathbb{1} + Ds - y + Bp\Big]\mathbb{1}^\top \\ &\quad \times (1./W) - 2\mathbb{1}(1./W)\Big[(E./W)\mathbb{1} - (E./W)^\top\mathbb{1} \\ &\quad + Ds - y + Bp\Big]^\top; \\ \nabla_p f(X) &= 2B^\top\Big[Ds - y + (E./W)\mathbb{1} - (E./W)^\top\mathbb{1} + Bp\Big] \\ &\quad + 2B^\top Bp. \end{aligned} \tag{24}$$

For any pairs of $X_1 = (s_1, E_1, p_1)^\top$, and $X_2 = (s_2, E_2, p_2)^\top$, some hand-waving shows that

$$\|\nabla f(X_1) - \nabla f(X_2)\|_F^2 \leq L_f^2 \left\| \begin{matrix} s_1 - s_2 \\ E_1 - E_2 \\ p_1 - p_2 \end{matrix} \right\|_F^2 \tag{25}$$

where we have employed the inequality $\langle A, B\rangle \leq \|A\|\|B\|_F$ extensively to obtain the upper bound. The last inequality and the definition for Lipschitz continuity verify the proposition. ■

Upon launching this, the application of APG is straightforward and we leave the detailed deductions to the appendix for completeness. With the knowledge of the sparse reconstruction coefficient $s$, the classification process is similar to [4]. For each class $i \in \{1, 2, \ldots, c\}$, let $\delta_i : \mathbb{R}^n \to \mathbb{R}^n$ be the characteristic function which selects the coefficients associated with the $i$-th class. For $s \in \mathbb{R}^n$, $\delta_i(s) \in \mathbb{R}$ is a new vector whose only nonzero entries are the entries in $s$ that are associated with class $i$. The whole optimization is presented in Algorithm 2. In implementation, we empirically set $\frac{\lambda_1}{r} = \lambda_2 = \lambda$ to considerably reduce the number of tunable parameters, where $r$ is the size of feature dimension. Moreover, we use the continuation technique to speed up overall convergence, i.e., we do not wait until convergence of the proximal problem to update the parameter $\mu$, but update it after each iteration by a multiplicative factor that is slightly less than 1. This technique has been widely applied in existing literature on APG, e.g., [16], [17], [24].

## IV. Discussion

There exists quite a few works related to the proposed RIASR method, including Robust Sparse Coding (RSC) [28] and Gabor Feature based Robust Sparse Coding (GSRC) [29]. Although the objectives of SRC and GSRC are similar as ours, the scopes and limitations to handle are different. For RSC, it assumes that the errors on each feature element are independently and identically distributed, while our algorithm reveals the relationship among them, that is ambiguity transfer and mis-detection transfer for errors among different histogram bins. For GSRC, it is based on the Gabor features, which effectively extract the image local directional features at multiple scales, while we target quantized histogram features, which are widely used for the superiority in encoding image structures or interest regions. Thus, RIASR is essentially different from

**Algorithm 2** Robust Image Analysis with SR Algorithm

**Input:** Let training samples be $D \in \mathbb{R}^{r\times n}$. The weighting matrix is $W \in [0,1]^{r\times r}$, a test sample is $y \in \mathbb{R}^r$, and weighting parameters is $\mu, \lambda$, let the decreasing factor be $\eta < 1$.
**Initialization:** $s_0 = s_{-1} = 0$; $E_0 = E_{-1} = 0$; $p_0 = p_{-1} = 0$; $b_0 = b_{-1} = -1$; $\bar{\mu} = 10^{-8}\mu_0$;
**While** *not converged* **do**
$Y_k^s = s_k + \frac{b_{k-1}-1}{b_k}(s_k - s_{k-1})$,
$Y_k^E = E_k + \frac{b_{k-1}-1}{b_k}(E_k - E_{k-1})$,
$Y_k^p = p_k + \frac{b_{k-1}-1}{b_k}(p_k - p_{k-1})$;
$O_k^s = 2D^\top DY_k^s + 2D^\top[-y + (Y_k^E./W)\mathbb{1} - (Y_k^E./W)^\top\mathbb{1} + BY_k^p]$;
$O_k^E = 2[(Y_k^E./W)\mathbb{1} - (Y_k^E./W)^\top\mathbb{1} + DY_k^s - y + BY_k^p]\mathbb{1}^\top(1./W) - 2\mathbb{1}(1./W)[(Y_k^E./W)\mathbb{1} - (Y_k^E./W)^\top\mathbb{1} + DY_k^s - y + BY_k^p]^\top$;
$O_k^p = 2B^\top[DY_k^s - y + (Y_k^E./W)\mathbb{1} - (Y_k^E./W)^\top\mathbb{1} + BY_k^p] + 2B^\top BY_k^p$;
$G_k^s = Y_k^s - \frac{1}{L_f}O_k^s$, $G_k^E = Y_k^E - \frac{1}{L_f}O_k^E$, $G_k^p = Y_k^p - \frac{1}{L_f}O_k^p$;
$s_{k+1} = T_{\frac{\mu}{L_f}}[G_k^s]$, $E_{k+1} = S_{\frac{\mu\lambda}{L_f}}[G_k^E]$, $p_{k+1} = S_{\frac{\mu\lambda}{L_f}}[G_k^p]$;
$b_{k+1} = \frac{1+\sqrt{4b_k^2+1}}{2}$, $\mu_{k+1} = \max(\eta\mu_k, \bar{\mu})$, $k = k+1$;
**end**
$s = s_{k+1}$
Compute the residuals $r_i(y) = \|y - D\delta_i(s)\|_2$, for $i = 1, \ldots, c$
**Output:** $\text{class}(y) = \arg\min_i r_i(y)$.

these methods and of its own perspective, so we do not plan to compare with these algorithms in experiment part.

## V. Experiments

In this section, we first demonstrate empirically the speed advantage of optimizing $E$ in Eqn. (12) using APG style proximal operator on matrix function over the naive translated Lasso problem (ref. Sec-III-A). Then we evaluate systematically the proposed Robust Image Analysis with SR framework (RIASR) for robust image analysis, on three benchmark datasets, Caltech-101 [30], Caltech-256 [31], Corel-5k [32], and CMU PIE [33], respectively.

### A. Standard Lasso Solution Versus APG Method on Matrix Function With $\ell_1$-Norm Regularizer

As discussed in Sec. III-A, it is prohibitive to optimize the matrix $E$ in Eqn. (12) with standard Lasso solving routines when the matrix dimension grows large. Instead the APG kind algorithm would be preferable due to the amendable closed-form solution to the proximal matrix subproblem.

We perform contrived simulations on toy data that is randomly generated, with the feature dimensionality $r$ ranging from 200 to 1800. Correspondingly the matrix $E$ is of $r \times r$; in comparison, the regressor matrix has a dimension of $r \times r^2$. We use two lasso solvers: Least Angle Regression (LARS) method [34] and block-coordinate descent (BCD) method[5] [35]. Figure 2 shows the comparison of running time among LARS, BCD and APG method, with the growth of dimensionality (hence the problem scale). From the figure, we can see when the feature dimension is small (=200), the running times of three methods are comparable; however, when the feature dimension increases through 800, the APG method turns out to be much faster than LARS and BCD solver. When the feature dimension goes up to near 1800, the running time of APG method is 1/75 of the cost by LARS and 1/10 of the cost by BCD solver.

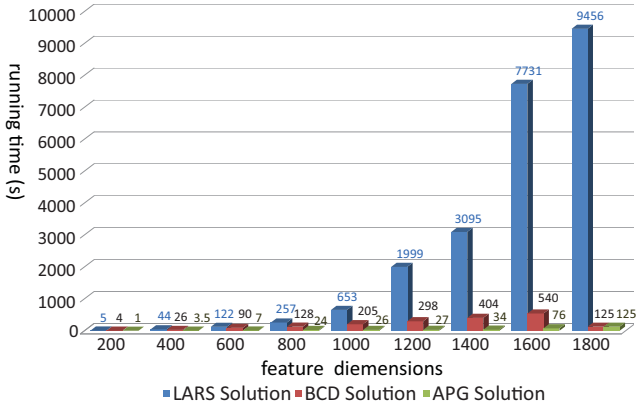


Fig. 2. Comparison of running times among the LARS solution (blue bar), BCD solution (red bar), and the APG solution (green bar) to optimize with respect to $E$. The $x$-axis indicates the feature dimension. The running times of two standard Lasso solutions quickly increase as the feature dimension grows, while that of the APG solution grows reasonably slowly.

We also empirically compare the overall running time for solving the RIASR problem over real data with: 1) alternating optimization based on LARS and BCD solvers; and 2) the APG algorithm proposed in our Algorithm 2. Here we just briefly report the running time on Caltech 101 [36], of which the feature dimension is set as 1000. When using LARS and BCD solvers, it respectively needs 3087s and 945s to process one image, while using the APG method, the running time decreases to 150s, which is 1/20 cost of that by LARS and 1/6 cost of that by BCD.

### B. Evaluation on Robust Image Analysis

The four benchmark datasets used in our experiments are Caltech-101 [36], Caltech-256 [31], Corel-5K [37] image datasets, and CMU PIE face dataset [38]. Caltech-101 is one of the most popular benchmark datasets for object recognition, containing 101 distinct categories and one background class with tens of samples per class. Caltech-256 is an extension of the Caltech-101 dataset. It consists of 256 object categories and contains from 80 to 827 images percategory. The total number of images is 30608. This dataset possesses larger intra-class variability than the Caltech-101 and thus is more challenging. Corel-5K is composed of 50 categories and each containing 100 images culled from the COREL image CDs. The CMU PIE (Pose, Illumination and Expression) database contains more than 40, 000 facial images of 68 people. The images were acquired across different poses, under variable

[5]The code is downloaded at http://www.mathworks.com/matlabcentral/fileexchange/25680-coordinate-descent-for-compressed-sensing.

illumination conditions, and with different facial expressions. We choose one near frontal pose C07, which includes 1629 images of 68 people in our experiment. Each image is aligned by fixing the location of the two eyes and normalized to 64-by-64 pixels. We choose sparse representation (SR) and support vector machines (SVM), which do not includes the process of eliminating the corruptions, as our baseline classifiers to show the performance for our proposed method in corrupted samples recognition.

*1) Object Recognition:* We first demonstrate the effectiveness of our RIASR framework on object recognition with Caltech-101, Caltech-256, and Corel-5k datasets. In Caltech-101 and Caltech-256, we randomly select 15 images for each category to form the training set, and the remaining 7614 and 26768 images for test respectively. In Corel-5k, 66 images out of 100 of each class are randomly selected as labeled samples, then the rest as unlabeled, leading to a train/test ratio of 3300 versus 1700.

For Caltech-101 and Caltech-256, we extract the features by following LLC method [39], which archives good performance on classification. Dense Scale-Invariant-Feature-Transform (SIFT) features [11] are extracted from densely located patches centered at every 4 pixels for each image and the size of the pixel is fixed as $16 \times 16$. We construct a visual word dictionary containing $K$ words via $K$-means clustering. We choose two values of $K$ for Caltech-101 and Caltech-256 respectively. One is set as 256, and the other one is 2048 for Caltech-101 and 4096 for Caltech-256, which is the same setting as state-of-the-arts reported by [39]. The corresponding feature size is 5376-dim for 256 codebook bases, 43008-dim for 2048 bases, and 86016-dim for 4096 bases. Due to memory limitation in our method, we use Marginal Fisher Analysis (MFA) [40] to reduce the feature dimension to 5000 for 21504-dim and 43008-dim features by feature selection. For Corel-5k, we extract 1000-dim dense SIFT feature to demonstrate the performance for the proposed algorithm on traditional Bag-of-Words features. All the features are $L_1$ normalized into a histogram form. We choose linear support vector machines (SVM), which is based on one-vs-all strategy, and sparse representation (SR) as our baselines.

In order to demonstrate the robustness of our method, we randomly add some corruptions into every test sample. The corruption-adding process simulates the two transfer processes that concern us in the current work. Hence there are two (concurrent) steps[6] in the process. To introduce ambiguity corruptions, we randomly select $x$ non-zero histogram-bins to corrupt, for $x \in \{0, \lfloor 0.1m \rfloor, \lfloor 0.2m \rfloor, \lfloor 0.3m \rfloor, \lfloor 0.4m \rfloor\}$, where $m$ is the number of non-zero histogram-bins, and $\lfloor \cdot \rfloor$ indicates the flooring operation. The discrete selection for $x$ simulates different levels of corruptions. The histogram bin that to be "mixed" with each bin in $\{x\}$ (we denote the set by $\{x\}$ for convenience) will be selected with probability in accordance with the weighting matrix $W$. The selected bin will then be added to a corruption value $z \sim U(0, y_i)$ that distributed uniformly between 0 and the original histogram value $y_i$. And the source bin will be deducted by the value $z$ accordingly. Moreover to simulate the mis-detection corruptions, we similarly selected different levels of source bins as above. We empirically assume the mis-detection transfer process does make the histogram denser in this experiment. Then, for the $i^{th}$ selected bin, we randomly decide whether the corruption decrease value $y_i$ or not. If "decrease", we assume the corruption value to be the smallest value $z$ of all other non-zero histogram-bins. Finally we add $(m-1) \times z$ to $y_i$, and minus $z$ from all the other non-zero ones. If "increase", we again generate a corruption value $z$ from the uniform distribution $U(0, y_i)$, and decrease the selected bin by $z$, and add $z$ to each of the other bins.

For parameter tuning, $\mu$ is tuned from $10^{-5}$ to $10^{-1}$, $\lambda$ is tune from 1 to 150, and $\eta$ is tuned from 0.7 to 0.995. We uniformly select 10 values form each of parameter range, then choose the highest one to fine tune. In Caltech-101, the parameters of RIASR are set as $\mu = 5^{-3}, \lambda = 5, \eta = 0.985$ when $K = 256$, and $\mu = 10^{-4}, \lambda = 5, \eta = 0.99$ when $K = 2048$. The parameter $C$ in linear-SVM is set as 50 both at $K = 256$ and $K = 2048$. $\epsilon$ in SR is set as 0.08. In Caltech-256, the parameters of RIASR are set as $\mu = 10^{-3}, \lambda = 50, \eta = 0.98$ when $K = 256$, and $\mu = 10^{-4}, \lambda = 50, \eta = 0.975$ when $K = 2048$. The parameter $C$ in linear-SVM is set as 50 both at $K = 256$ and $K = 2048$. $\epsilon$ in SR is set as 0.1 In Corel-5k, the parameters of RIASR are set as $\mu = 10^{-3}, \lambda = 100, \eta = 0.7$, and that of linear-SVM is set as $C = 100$. In SR, $\epsilon$ is set as 0.002.

[6]Due to the linearly additive nature of corruptions caused by these two processes as discussed previously, there is no difference by implementing the concurrent corruptions sequentially as we describe here.

TABLE I
RECOGNITION ACCURACY (%) COMPARISON AGAINST DIFFERENT CORRUPTION RATES ON CALTECH-101 DATASET. THE FIFTH ROW SHOWS THE RESULTS FROM STATE-OF-THE-ARTS ALGORITHM

| Algorithm | | 0% | 10% | 20% | 30% | 40% |
|---|---|---|---|---|---|---|
| K=256 | linear-SVM | **60.25** | 54.78 | 50.44 | 46.34 | 40.58 |
| | SR | 50.36 | 46.23 | 44.1 | 42.08 | 40.2 |
| | SIASR | 58.28 | **55.85** | **54.44** | **53.68** | **52.11** |
| K=2048 | linear-SVM | **67.35** [39] | 64.37 | 60.28 | 57.33 | 54.75 |
| | SR | 57.52 | 53.81 | 50.06 | 48.4 | 46.12 |
| | SIASR | 66.39 | **65.23** | **64.43** | **63.28** | **61.87** |

TABLE II
RECOGNITION ACCURACY (%) COMPARISON AGAINST DIFFERENT CORRUPTION RATES ON CALTECH-256 DATASET. THE FIFTH ROW SHOWS THE RESULTS FROM STATE-OF-THE-ARTS ALGORITHM

| Algorithm | | 0% | 10% | 20% | 30% | 40% |
|---|---|---|---|---|---|---|
| K=256 | linear-SVM | **10.97** | 8.48 | 6.75 | 5.43 | 5.28 |
| | SR | 8.4 | 7.28 | 5.84 | 5.22 | 5.22 |
| | SIASR | 10.53 | 9.32 | 8.84 | 8.42 | 7.23 |
| K=4096 | linear-SVM | **34.36** [39] | **31.52** | 27.33 | 25.26 | 23.15 |
| | SR | 25.54 | 22.72 | 19.80 | 18.14 | 16.82 |
| | SIASR | 32.92 | 31.05 | **29.87** | **29.05** | **27.33** |

TABLE III
RECOGNITION ACCURACY (%) COMPARISON AGAINST DIFFERENT CORRUPTION RATES ON COREL-5 K DATASET

| Algorithm | 0% | 10% | 20% | 30% | 40% |
|---|---|---|---|---|---|
| Linear-SVM | 62.44 | 57.52 | 56.83 | 50.26 | 48.32 |
| SR | 57.57 | 49.58 | 47.36 | 42.58 | 40.31 |
| SIASR | **64.30** | **63.92** | **63.14** | **62.56** | **61.2** |

Table I shows the classification performance against corruption rate on Caltech-101. For clean test data, the accuracy of linear-SVM is (67.35%) when $K = 2048$ and 60.25% when $K = 256$, while our results are slight lower, that is 66.39% (1% lower) and 58.28% (2% lower) respectively. However, the proposed method is more robust on corrupted data. From clean data to 40% corrupted data, the decrease of accuracy for RIASR is 6.17% when $K = 256$, and 4.52% when $K = 2048$, while whose for SR are 10.16% and 11.4%, whose for linear-SVM are 19.67% and 12.6%. Table I clearly shows that the decrease of accuracy in RIASR (5.17%) is much less than those in linear-SVM (19.7%) and SR (10.2%) when the corruption rate for test data reaches 40%.

Table II shows the classification performance against corruption rate on Caltech-256. The result is similar to that on Caltech-101. RIASR achieves more robust performance when the corruption rate reaches to 40% than SR and linear-SVM.

We also demonstrate the performance for our method with other histogram feature, SIFT-Bows, on Corel-5k. As shown in Table III, the accuracy of RIASR reaches 64.3% for clean test data, which is higher than that of linear-SVM (62.44%), and that of SR (57.57%) For the corrupted test data, the figure also shows that the decrease of accuracy in RIASR (3.1%) is much less than those in linear-SVM (14.12%) and SR (17.26%).

*2) Face Recognition:* In this experiment, we randomly select 10 images in every category as the training samples, and hence split the dataset into 680 training samples and 1012 test samples.

Whereas the corruptions in previous experiments are simple and only occurs at the BoW level, this time we make a more complicated scenario whereby corruptions occur in multiple levels of quantization and the relatively clean data are unknown to us (recall in the last experiment, we artificially introduced corruptions to histograms that we assumed to contain no corruptions). To this end, we corrupt a certain percentage of randomly chosen pixels from each of the test images, replacing their values with iid samples from a uniform distribution. The corrupted pixels are randomly chosen for each test image and the locations are unknown to the computer. We vary the percentage of corrupted pixels from 0% to 40%.

We extract two kinds of features to demonstrate the performance of our algorithm, one is pixel-based raw feature, the other one is LBP feature. All the images are cropped with dimension $64 \times 64$, and converted to grayscale. For pixel-based raw feature, each feature is normalized into histogram feature. For LBP feature, the $\mathrm{LBP}^{u2}_{8,1}$ [12] raw feature is extracted for every pixel. Then the image is divided into $7 \times 7$ rectangular regions, and a histogram is computed independently within every region. At last, these 49 ($=7 \times 7$) histograms are merged into one histogram by lining them up with different weights. The extraction process and weights selection are according to [41]. The parameters in this experiment are set as $\mu = 10^{-5}, \lambda = 100, \eta = 0.98$.

TABLE IV
DETAILED CLASSIFICATION ACCURACIES (%) AGAINST DIFFERENT CORRUPTION RATES (RIASR VERSUS SR ON PIE)

| Algorithm | 0% | 10% | 20% | 30% | 40% |
|---|---|---|---|---|---|
| Raw + SR | 94.35 | 91.81 | 88.64 | 83.42 | 75.38 |
| Raw + SIASR | **96.34** | **95.85** | **93.22** | **91.68** | **88.04** |
| LBP + SR | 82.93 | 77.69 | 68.42 | 41.57 | 15.46 |
| LBP + SIASR | **92.20** | **90.02** | **85.34** | **78.38** | **69.87** |

TABLE V
RECOGNITION ACCURACIES (%) AGAINST DIFFERENT NUMBERS OF TRAINING SAMPLES ON PIE DATASET. THE FIRST COLUMN LISTS THE ALGORITHMS TO EVALUATE, AND OTHER COLUMNS SHOW THE AVERAGE VALUES AND STANDARD DEVIATIONS AGAINST DIFFERENT NUMBERS OF TRAINING SAMPLES FROM 10 RANDOM SPLITS OF THE DATASETS

| Algorithm | 5 | 10 | 15 | 20 |
|---|---|---|---|---|
| SR | 71.02(±1.66) | 84.25(±1.41) | 88.80(±1.32) | 91.10(±1.01) |
| SIASR | 82.93(±1.67) | 92.20(±1.31) | 95.23(±0.82) | 96.56(±0.72) |

Table IV presents the classification performances against different corruption rates on CMU PIE dataset when the number of training data is set as 10. We demonstrate two kinds of features, LBP and raw feature, and two kinds of algorithms, SR and our proposed one. As the corruption rate increases from 0% to 40%, the degradation for the accuracy of SR is much sharper than that of RIASR both on LBP and raw feature. We also experiment on the original PIE images, which are not artificially corrupted, to verify the classification performance of our algorithm in practice. We randomly select $n_l \in \{5, 10, 15, 20\}$ images for training, and the rest for test. The reported mean and standard deviation of the recognition accuracy are estimated over 10 random splits. Table V shows the recognition accuracies against different numbers of training samples on PIE dataset. From the table, we can see SIASR algorithm performs considerably better than the original SR algorithm.

## VI. CONCLUSION

In this paper, we have proposed the Robust Image Analysis with Sparse Representation (RIASR) algorithm, which not only remedies the corruptions caused by ambiguity transfer and mis-detection transfer in quantized visual features, but is also verified to be efficient in optimization. The experiments on three benchmark object recognition datasets, Caltech-101, Caltech-256 and Corel-5k, and one popular face recognition dataset, CMU PIE, have demonstrated strongly the practical effectiveness and efficiency, both on the contrived scenarios and the real. This is just the start of this line of work that has a strong practical flavor, e.g., we can extend our basic ideas from dealing with quantization corruptions to other processes that cause corruptions, e.g. feature extraction process. In this way, the feature extraction will be more

robust and accurate. From another perspective, our framework only considers the corruptions contaminating the test sample with clean training samples for the SR reconstruction. It would be interesting to extend to a more realistic model, where the training samples also contain corruptions. The analysis would be more challenging and rewarding. Our last experiment on the original CMU PIE data has produced encouraging results in this line. And some of the recent works in the signal processing community also share the same spirit, e.g., [42].

## APPENDIX

### A. Proof of Corruption Additivity

In this subsection, we will prove two corruptions, ambiguity transfer and mis-detected transfer, are linear additive. For the sake of simplicity, the ambiguity transfer is assumed to occur from $i$-th bin to $j$-th, and the mis-detected transfer is from $i$-th bin to any others.

For a certain image, let clean data be $x = \{x_1, x_2, \ldots, x_i, \ldots, x_j, \ldots, x_r\}^T$. For ambiguity transfer, let the amount of shift in normalized histogram values from $i$-th bin to $j$-th bin be $k_{ji}$. For mis-detection transfer, let the "mis-detection vector" $p = \{p_1, \ldots, p_r\}^T$, where the element $p_i$ indicates the unnormalized amount of corruption in $i$-th bin. So the corrupted data $y = \{y_1, \ldots, y_i, \ldots, y_j, \ldots, y_r\}^T$ are equal to

$$y_1 = \frac{x_1 + p_1}{1 + \sum p}, \ldots, y_i = \frac{x_i + p_i - k_{ji}}{1 + \sum p}, \ldots,$$
$$y_j = \frac{x_j + p_j + k_{ji}}{1 + \sum p}, \ldots, y_r = \frac{x_r + p_r}{1 + \sum p}.$$

Then original clean data $x = \{x_1, \ldots, x_i, \ldots, x_j, \ldots, x_r\}^T$ can be calculated as

$$x_1 = (1+\sum p)y_1 - p_1, \ldots, \quad x_i = (1+\sum p)y_i - p_i + k_{ji},$$
$$x_j = (1+\sum p)y_j - p_j - k_{ji}, \ldots, \quad x_r = (1+\sum p)y_r - p_r.$$

Then, the overall corruptions $\mathcal{E} = \{\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_i, \ldots, \mathcal{E}_j, \ldots, \mathcal{E}_n\}$ is

$$\mathcal{E}_1 = y_1 - x_1 = -\sum p y_1 + p_1, \ldots,$$
$$\mathcal{E}_i = y_i - x_i = -\sum p y_i + p_i - k_{ji}, \ldots,$$
$$\mathcal{E}_j = y_j - x_j = -\sum p y_j + p_j + k_{ji}, \ldots,$$
$$\mathcal{E}_r = y_r - x_r = -\sum p y_r + p_r.$$

Since $\mathcal{E}_M = \sum py + p$, which is deduced in Eqn. (3), and $\mathcal{E}_A = K\mathbb{1} - K^\top\mathbb{1}$, where $K$ is matrix with $k_{ij} = 0$, other elements are equal to 0. Therefore, $\mathcal{E} = \mathcal{E}_M + \mathcal{E}_A$, which is linear additive.

### B. Proof of Algorithm Convergence

We present the detailed derivation of the iterative process with APG to optimize the problem in Eqn. (12). Specifically, we derive the rules to update $s$, $E$ and $p$ in each iteration. From Eqn. (16), (17), and (24), we get the update for $X$ as

$$\begin{aligned} X_{k+1} &= \arg\min_X Q(X, Y_k) \\ &= \arg\min_X f(Y_k) + \langle \nabla f(Y_k), X - Y_k \rangle \\ &\quad + \frac{L_f}{2}\|X - Y_k\|_F^2 + \mu g(X) \end{aligned} \tag{26}$$

where $X = (s, E, p)^\top$ and $Y_k = \left(Y_k^s, Y_k^E, Y_k^p\right)^\top$. Observing the separability of each additive term w.r.t. components of $X$ in Eqn. (26), we have the following updating equations.

*1) Optimizing $s$:*

$$s_{k+1} = \arg\min_{s\geq 0} \tfrac{L_f}{2}\|s - Y_k^s\|_F^2 + \mu\|s\|_1 + \left\langle J, s - Y_k^s \right\rangle$$

where $J = 2D^T DY_k^s + 2D^\top[-y + (Y_k^E./W)\mathbb{1} - (Y_k^E./W)^\top\mathbb{1} + BY_k^p]$. Let $O_k^s = 2D^\top DY_k^s + 2D^T[-y + (Y_k^E./W)\mathbb{1} - (Y_k^E./W)^T\mathbb{1} + BY_k^p]$, we then have

$$s_{k+1} = \arg\min_{s\geq 0} \frac{L_f}{2}\left\| s - Y_k^s + \frac{1}{L_f}O_k^s \right\|_F^2 + \mu\|s\|_1 \tag{27}$$

where above we have dropped constant terms due to the fixed $E$ and $p$. It follows immediately from Proposition 3.1

$$s_{k+1} = T_{\frac{\mu}{L_f}}\left[ Y_k^s - \frac{1}{L_f}O_k^s \right]. \tag{28}$$

*2) Optimizing $E$:* Similar to the deduction procedure of $s$, we have

$$\begin{aligned} E_{k+1} &= \arg\min_E Q(E, Y_k) \\ &= \arg\min_E \frac{L_f}{2}\left\| E - Y_k^E + \frac{1}{L_f}O_k^E \right\|_F^2 + \mu\lambda_1\|E\|_1 \end{aligned}$$

where we have made similar substitution using $O_k^E$ for the gradient term. The updating equation follows from Proposition 3.1

$$E_{k+1} = S_{\frac{\mu\lambda_1}{L_f}}\left[ Y_k^E + \frac{1}{L_f}O_k^E \right]. \tag{29}$$

*3) Optimizing $p$:*

$$\begin{aligned} p_{k+1} &= \arg\min_p Q(p, Y_k) \\ &= \arg\min_p \frac{L_f}{2}\left\| p - Y_k^p + \frac{1}{L_f}O_k^p \right\|_F^2 + \mu\lambda_2\|p\|_1. \end{aligned} \tag{30}$$

We figure out the solution as

$$p_{k+1} = S_{\frac{\mu\lambda_2}{L_f}}\left[ Y_k^p + \frac{1}{L_f}O_k^p \right]. \tag{31}$$

## REFERENCES

[1] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[2] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, Jan. 2008.

[3] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.
[4] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2008.
[5] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1794–1801.
[6] B. Bao, T. Li, and S. Yan, "Hidden-concept driven image decomposition toward semi-supervised multi-label image annotation," in *Proc. Int. Conf. Internet Multimedia Comput. Service*, 2009, pp. 17–24.
[7] A. Bruckstein, D. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, 2009.
[8] D. George and J. Hawkins, "Toward a mathematical theory of cortical micro-circuits," *PLoS Comput. Biol.*, vol. 5, no. 10, pp. 1–8, 2009.
[9] D. Graham and D. Field, "Sparse coding in the neocortex," *Evol. Nervous Syst.*, vol. 3, no. 2, pp. 181–187, 2006.
[10] R. Szeliski, *Computer Vision: Algorithms and Applications*, 1st ed. Berlin, Germany: Springer-Verlag, 2011.
[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[12] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
[13] N. Dalal, B. Triggs, I. Rhone-Alps, and F. Montbonnot, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 886–893.
[14] P. Zhao and B. Yu, "On model selection consistency of lasso," *J. Mach. Learn. Res.*, vol. 7, pp. 2541–2567, Nov. 2006.
[15] S. Becker, J. Bobin, and E. Candès, "NESTA: A fast and accurate first-order method for sparse recovery," *SIAM J. Imag. Sci.*, vol. 4, no. 1, pp. 1–39, 2009.
[16] P. Tseng, "On accelerate proximal gradient methods for convex-concave optimization," *SIAM J. Optim.*, to be published.
[17] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
[18] E. Amaldi and V. Kann, "On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems," *Theor. Comput. Sci.*, vol. 209, nos. 1–2, pp. 237–260, 1998.
[19] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
[20] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal Stat. Soc. B (Methodol.)*, vol. 58, no. 1, pp. 267–288, 1996.
[21] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *J. Res. Statist. Soc. B.*, vol. 58, no. 1, pp. 267–288, Sep. 2010.
[22] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $o(1/k^2)$," *Sov. Math. Doklady*, vol. 27, no. 3, pp. 372–376, 1983.
[23] N. Yu, "Smooth minimization of non-smooth functions," *Math. Program.*, vol. 103, no. 1, pp. 127–152, 2005.
[24] K. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems," *Pacific J. Optim.*, vol. 6, pp. 615–640, Mar. 2010.
[25] G. Zhu, S. Yan, and Y. Ma, "Image tag refinement towards low-rank, content-tag prior and error sparsity," in *Proc. Int. Conf. Multimedia*, 2010, pp. 461–470.
[26] D. Donoho, "De-noising via soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.
[27] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[28] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 625–632.
[29] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with gabor occlusion dictionary," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 448–461.
[30] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," in *Proc. IEEE CVPR Workshop Generat.-Model Based Vis.*, Jun. 2004, p. 178.
[31] A. H. G. Griffin and P. Perona, "Caltech-256 object category dataset," Dept. Comput. Sci., California Inst. Technology, Pasadena, Tech. Rep. CNS-TR-2007-001, Apr. 2007.
[32] P. Duygulu, K. Barnard, J. D. Freitas, and D. Forsyth, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," in *Proc. 7th Eur. Conf. Comput. Vis.*, 2002, pp. 349–354.
[33] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proc. IEEE Int. 5th Autom. Face Gesture Recognit. Conf.*, May 2002, pp. 46–51.
[34] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Stat.*, vol. 32, no. 2, pp. 407–451, 2004.
[35] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani, "Pathwise coordinate optimization," *Ann. Appl. Stat.*, vol. 1, no. 2, pp. 302–332, 2007.
[36] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Comput. Vis. Image Understand.*, vol. 106, no. 1, pp. 59–70, 2007.
[37] J. Yuan, J. Li, and B. Zhang, "Exploiting spatial context constraints for automatic image region annotation," in *Proc. 15th ACM Int. Conf. Multimedia*, 2007, pp. 595–604.
[38] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.
[39] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. Comput. Vis. Pattern Recogn.*, 2010, pp. 3360–3367.
[40] S. Yan, D. Xu, B. Zhang, H. J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.
[41] T. Ahonen and A. Hadid, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.
[42] H. Zhu, G. Leus, and G. B. Giannakis, "Sparsity-cognizant total least-squares for perturbed compressive sampling," *CoRR*, vol. 59, pp. 1–30, Aug. 2010.

**Bing-Kun Bao** received the Ph.D. degree in control theory and control application from the Department of Automation, University of Science and Technology of China, Hefei, China, in 2009.

She was a Research Engineer of electrical and computer engineering with the National University of Singapore, Singapore, from 2009 to 2011. She is currently a Post-Doctoral Researcher with the Institute of Automation, Chinese Academy of Science, Beijing, China, and a Researcher with the China-Singapore Institute of Digital Media, Singapore.

Dr. Bao was a recipient of the Best Paper Award from ICIMCS'09. She was the Special Session Organizer and Technical Program Committee Member of MMM'13.

**Guangyu Zhu** received the M.S. and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 2003 and 2008, respectively.

He was a Senior Research Faculty Member with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, and a Research Scientist with NEC Laboratories America, Princeton, NJ. His current research interests include image and video processing, multimedia content analysis, computer vision and pattern recognition, and machine learning.

Dr. Zhu was a Technical Program Committee Member of MMM'07, CIVR'10, ICIP'10, and ICIMCS'10. He was the Special Session Chair of ACM ICIMCS 2009 and a Reviewer of the IEEE TRANSACTIONS OF MULTIMEDIA and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, *Multimedia Systems*, and *Pattern Recognition*.

**Jialie Shen** is currently an Assistant Professor of information systems with the School of Information Systems, Singapore Management University, Singapore. His recent research has been published or is forthcoming in leading journals and international conferences, including ACM SIGIR, ACM Multimedia, ACM SIGMOD, CVPR, ICDE, WWW, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the *ACM Multimedia Systems Journal*, the *ACM Transactions on Internet Technology*, and the *ACM Transactions on Information Systems*. His current research interests include information retrieval, multimedia systems, economic-aware media analysis, and statistical machine learning.

Prof. Shen is the Chair, a PC member, a Reviewer, or a Guest Editor of several leading information systems journals and conferences. He is an Associate Editor of the *International Journal of Image and Graphics*.

**Shuicheng Yan** (M'08–SM'09) is currently an Assistant Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, and the Founding Lead of the Learning and Vision Research Group (http://www.lv-nus.org). He has authored or co-authored nearly 300 technical papers over a wide range of research topics. His current research areas include computer vision, multimedia and machine learning.

Dr. Yan was a recipient of the Best Paper Award at PCM'11, ACM MM'10, ICME'10, and ICIMCS'09, the Winner Prize of the Classification Task at both PASCAL VOC'10 and PASCAL VOC'11, the Honorable Mention Prize of the Detection Task at PASCAL VOC'10, the TCSVT Best Associate Editor (BAE) Award in 2010, the Young Faculty Research Award in 2010, the Singapore Young Scientist Award in 2011, the NUS Young Researcher Award in 2012, and the Best Student Paper Award for co-authored paper at PREMIA'09, PREMIA'11, and PREMIA'12. He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and the *ACM Transactions on Intelligent Systems and Technology*, and is a Guest Editor of the special issues for TMM and CVIU.