Sparsity Control for Robust Principal Component Analysis

Gonzalo Mateos and Georgios B. Giannakis
Dept. of ECE, University of Minnesota
200 Union St. SE, Minneapolis, MN 55455, USA
Emails: {mate0058,georgios}@ece.umn.edu

***Abstract*—Principal component analysis (PCA) is widely used for high-dimensional data analysis, with well-documented applications in computer vision, preference measurement, and bioinformatics. In this context, the fresh look advocated here permeates benefits from variable selection and compressive sampling, to robustify PCA against outliers. A least-trimmed squares estimator of a low-rank component analysis model is shown closely related to that obtained from an ℓ_0-(pseudo)norm-regularized criterion encouraging *sparsity* in a matrix explicitly modeling the outliers. This connection suggests efficient (approximate) solvers based on convex relaxation, which lead naturally to a family of robust estimators subsuming Huber's optimal M-class. Outliers are identified by tuning a regularization parameter, which amounts to controlling the sparsity of the outlier matrix along the whole *robustification* path of (group)-Lasso solutions. Novel algorithms are developed to: i) estimate the low-rank data model both robustly and adaptively; and ii) determine principal components robustly in (possibly) infinite-dimensional feature spaces. Numerical tests corroborate the effectiveness of the proposed robust PCA scheme for a video surveillance task.**

I. Introduction

Principal component analysis (PCA) is the workhorse of high-dimensional data analysis and dimensionality reduction, with numerous applications in statistics, engineering, and the social sciences; see, e.g., [7]. Nowadays ubiquitous e-commerce sites, the World Wide Web, and urban traffic surveillance systems generate massive volumes of data. As a result, the problem of extracting the most informative – yet low-dimensional – structure from high-dimensional datasets is of paramount importance [5]. In this direction, PCA provides least-squares (LS) optimal linear approximations to a data set in $\mathbb{R}^p$ of any rank $q \leq p$. The desired linear subspace is efficiently obtained from the q dominant eigenvectors of the sample data covariance matrix [7].

Along with data that adhere to postulated models, present in large volumes of data are those that do not (outliers) [12]. Unfortunately, LS is known to be very sensitive to outliers [12], [6], and this undesirable property is inherited by PCA as well [7]. Early efforts towards robustifying PCA have relied on robust estimates of the data covariance matrix; see, e.g., [1]. Statistical physics have been applied to robust PCA in [15], while a method building on M-estimators was put forth in [3]. Recently, polynomial-time algorithms with performance guarantees were developed for low-rank matrix recovery in the presence of sparse errors [2]. This pertains to an idealized robust PCA problem, since those entries not affected by outliers are assumed to be observed without errors. Moreover, it is not possible to determine principal components from the low-rank matrix recovered.

In the present paper, a robust PCA scheme is developed which requires minimal assumptions on the outlier model. A natural least-trimmed squares (LTS) PCA estimator is first shown closely related to an estimator obtained from an ℓ_0-(pseudo)norm-regularized criterion, adopted to fit a low-rank component analysis model that explicitly incorporates an unknown *sparse* vector of outliers per datum. As in compressive sampling [14], efficient (approximate) solvers are obtained by surrogating the ℓ_0 norm of the outlier matrix with its closest convex approximant. This leads naturally to an M-type PCA estimator which subsumes Huber's optimal choice as a special case [4]. Unlike Huber's formulation though, results here are not confined to an outlier contamination model. A tunable parameter controls the sparsity of the estimated matrix, and the number of outliers as a byproduct. Hence, effective methods to select this parameter are of paramount importance, and systematic approaches are pursued by efficiently exploring the whole *robustifaction* path of (group-)Lasso solutions [5], [17]. In this sense, the method here capitalizes on but *is not limited to* sparse settings where outliers are sporadic, since one can examine all sparsity levels along the robustification path.

Novel robust algorithms are developed to: i) adaptively estimate the low-rank data model as new data comes in; and ii) determine principal components in (possibly) infinite-dimensional feature spaces, thus robustifying kernel PCA as well [13]. Numerical tests on both real and synthetic data demonstrate the effectiveness of the novel methods.

Notation: Operators $(\cdot)'$, $\text{tr}(\cdot)$, and $\text{med}(\cdot)$ will denote transposition, matrix trace, and median, respectively; vector $\text{diag}(\mathbf{M})$ collects the diagonal elements of $\mathbf{M}$. The ℓ_p norm of vector $\mathbf{x}$ is $\|\mathbf{x}\|_p := \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$ for $p \geq 1$; and $\|\mathbf{M}\|_F := \sqrt{\text{tr}(\mathbf{M}\mathbf{M}')}$ is the matrix Frobenious norm. The $p \times p$ identity matrix will be represented by $\mathbf{I}_p$, while $\mathbf{0}_p$ ($\mathbf{1}_p$) will denote the $p \times 1$ vector of all zeros (ones), and $\mathbf{0}_{p\times q} := \mathbf{0}_p\mathbf{0}_q'$.

II. Robustifying PCA

Consider the classical PCA problem [7], in which a set of data $\mathcal{T} := \{\mathbf{x}_n\}_{n=1}^N$ in the p-dimensional Euclidean *input*

† Work in this paper was supported by the NSF grants CCF-0830480, 1016605, and ECCS-0824007, 1002180.

978-1-4244-9721-8/10/$26.00 ©2010 IEEE

space is given, and the goal is to find the best q-rank ($q \leq p$) linear approximation to the data in $\mathcal{T}$. One approach to solving this problem, is to adopt a low-rank (component analysis) model

$$\mathbf{x}_n = \mathbf{m} + \mathbf{U}\mathbf{s}_n + \mathbf{e}_n, \quad n = 1, \ldots, N \tag{1}$$

where $\mathbf{m} \in \mathbb{R}^p$ is a location (mean) vector; matrix $\mathbf{U} \in \mathbb{R}^{p \times q}$ has orthonormal columns spanning the signal subspace; $\{\mathbf{s}_n\}_{n=1}^N$ are the so-termed *principal components*, and $\{\mathbf{e}_n\}_{n=1}^N$ are zero-mean i.i.d. random errors. The unknowns in (1) can be collected in $\mathcal{U} := \{\mathbf{m}, \mathbf{U}, \{\mathbf{s}_n\}_{n=1}^N\}$, and they are estimated via LS. The resulting estimates are $\hat{\mathbf{m}} = \sum_{n=1}^N \mathbf{x}_n / N$ and $\hat{\mathbf{s}}_n = \hat{\mathbf{U}}'(\mathbf{x}_n - \hat{\mathbf{m}})$, $n = 1, \ldots, N$; while $\hat{\mathbf{U}}$ is given by the q-dominant right singular vectors of the $N \times p$ data matrix $\mathbf{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]'$ [5, p. 535]. Note that the principal components (entries of) $\mathbf{s}_n$ are the projections of the centered data points onto the signal subspace. Equivalently, PCA can be formulated to optimize maximum variance, or, minimum reconstruction error criteria; see, e.g., [7].

Given training data in $\mathcal{T}$ possibly contaminated with outliers, the goal here is to develop a robust estimator of $\mathcal{U}$ that requires minimal assumptions on the outlier model. Building on LTS regression [12], the desired robust estimate $\hat{\mathcal{U}}_{LTS} := \{\hat{\mathbf{m}}, \hat{\mathbf{U}}, \{\hat{\mathbf{s}}_n\}_{n=1}^N\}$ can be obtained as the minimizer of the following LTS PCA estimate

$$\hat{\mathcal{U}}_{LTS} := \arg\min_{\mathcal{U}} \sum_{n=1}^{\nu} r_{[n]}^2(\mathcal{U}) \tag{2}$$

where $r_{[n]}^2(\mathcal{U})$ is the n-th order statistic among the squared residual norms $r_1^2(\mathcal{U}), \ldots, r_N^2(\mathcal{U})$, and $r_n(\mathcal{U}) := \|\mathbf{x}_n - \mathbf{m} - \mathbf{U}\mathbf{s}_n\|_2$. The so-termed *coverage* ν determines the breakdown point of the LTS PCA estimator [12], since $N - \nu$ residuals are not present in (2). Even though (2) is nonconvex, existence of a minimizer $\hat{\mathcal{U}}_{LTS}$ can be established as follows: i) for each subset of $\mathcal{T}$ with cardinality ν (there are $\binom{N}{\nu}$ such subsets), solve the corresponding PCA problem to obtain a candidate estimator per subset; and ii) pick $\hat{\mathcal{U}}_{LTS}$ as the one among all $\binom{N}{\nu}$ candidates with the least cost. This solution procedure is combinatorially complex, and thus intractable except for small sample sizes N. Algorithms to obtain approximate LTS solutions in linear regression are available; see e.g., [12].

Instead of discarding large residuals, the alternative approach here explicitly accounts for outliers in the low-rank data model. To this end, consider the vector variables $\{\mathbf{o}_n\}_{n=1}^N$ one per training data point, which take the value $\mathbf{o}_n \neq \mathbf{0}_p$ whenever datum n is an outlier, and $\mathbf{o}_n = \mathbf{0}_p$ otherwise. This leads to the outlier-aware factor analysis model

$$\mathbf{x}_n = \mathbf{m} + \mathbf{U}\mathbf{s}_n + \mathbf{o}_n + \mathbf{e}_n, \qquad n = 1, \ldots, N \tag{3}$$

where the $\mathbf{o}_n$ can be deterministic or random with unspecified distribution. In the *under-determined* linear system of equations (3), both $\mathcal{U}$ as well as the $N \times p$ matrix $\mathbf{O} := [\mathbf{o}_1, \ldots, \mathbf{o}_N]'$ are unknown. The percentage of outliers dictates the degree of *sparsity* (number of all-zero rows) in $\mathbf{O}$. Sparsity control will prove instrumental in efficiently estimating $\mathbf{O}$, rejecting outliers as a byproduct, and consequently arriving at a *robust* estimator of $\mathcal{U}$. A natural criterion for controlling outlier sparsity is to seek the estimator

$$\begin{aligned}\{\hat{\mathcal{U}}, \hat{\mathbf{O}}\} = \; & \arg\min_{\mathcal{U}, \mathbf{O}} \|\mathbf{X} - \mathbf{1}_N \mathbf{m}' - \mathbf{S}\mathbf{U}' - \mathbf{O}\|_F^2 + \lambda_0 \|\mathbf{O}\|_0 \\ & \text{s.t. } \mathbf{U}'\mathbf{U} = \mathbf{I}_q \end{aligned} \tag{4}$$

where $\mathbf{S} := [\mathbf{s}_1, \ldots, \mathbf{s}_N]' \in \mathbb{R}^{N \times q}$, and $\|\mathbf{O}\|_0$ denotes the nonconvex ℓ_0-(pseudo)norm that is equal to the number of nonzero rows of $\mathbf{O}$. Vector (group) sparsity in the rows $\hat{\mathbf{o}}_n$ of $\hat{\mathbf{O}}$ can be directly controlled by tuning the parameter $\lambda_0 \geq 0$.

As with compressive sampling and sparse modeling schemes that rely on the ℓ_0-norm [14], the robust PCA problem (4) is NP-hard. In addition, the sparsity-controlling estimator (4) is intimately related to LTS PCA, as asserted next [11].

Proposition 1: *If* $\{\hat{\mathcal{U}}, \hat{\mathbf{O}}\}$ *minimizes* (4) *with* λ_0 *chosen such that* $\|\hat{\mathbf{O}}\|_0 = N - \nu$, *then* $\hat{\mathcal{U}}_{LTS} = \hat{\mathcal{U}}$ *in* (2).

The importance of Proposition 1 is threefold. First, it formally justifies model (3) and its estimator (4) for robust PCA, in light of the well documented merits of LTS [12]. Second, it further solidifies the connection between sparsity-aware learning and robust estimation. Third, problem (4) lends itself naturally to efficient (approximate) solvers based on convex relaxation, the subject dealt with next.

III. Sparsity Controlling Outlier Rejection

Recall that the row-wise ℓ_2-norm sum $\|\mathbf{B}\|_{2,r} := \sum_{n=1}^N \|\mathbf{b}_n\|_2$ of matrix $\mathbf{B} := [\mathbf{b}_1, \ldots, \mathbf{b}_N]' \in \mathbb{R}^{N \times p}$ is the closest convex approximation of $\|\mathbf{B}\|_0$. This property provides the motivation to relax problem (4) to

$$\begin{aligned}& \min_{\mathcal{U}, \mathbf{O}} \|\mathbf{X} - \mathbf{1}_N \mathbf{m}' - \mathbf{S}\mathbf{U}' - \mathbf{O}\|_F^2 + \lambda_2 \|\mathbf{O}\|_{2,r} \\ & \text{s.t. } \mathbf{U}'\mathbf{U} = \mathbf{I}_q. \end{aligned} \tag{5}$$

The nondifferentiable ℓ_2-norm regularization term controls row-wise (vector) sparsity on the estimator of $\mathbf{O}$, a property that has been exploited in diverse problems in engineering, statistics, and machine learning [5]. A noteworthy representative is the group Lasso [17], a popular tool for joint estimation and selection of grouped variables in linear regression.

It is pertinent to ponder on whether problem (5) still has the potential of providing robust estimates $\hat{\mathcal{U}}$ in the presence of outliers. The answer is positive, since it is possible to show that (5) is equivalent to an M-type estimator [11]

$$\min_{\mathcal{U}} \sum_{n=1}^N \rho_v(\mathbf{x}_n - \mathbf{m} - \mathbf{U}\mathbf{s}_n) \tag{6}$$

where $\rho_v : \mathbb{R}^p \to \mathbb{R}$ is a vector extension to Huber's convex loss function [6]; see also [8]

$$\rho_v(\mathbf{r}) := \begin{cases} \|\mathbf{r}\|_2^2, & \|\mathbf{r}\|_2 \leq \lambda_2/2 \\ \lambda_1 \|\mathbf{r}\|_2 - \lambda_2^2/4, & \|\mathbf{r}\|_2 > \lambda_2/2 \end{cases}. \tag{7}$$

Previous efforts towards robustifying linear regression have pointed out the equivalence between M-type estimators and ℓ_1-norm regularized regression [4]. However, they have not recognized the connection to LTS via convex relaxation of

(4). Here, the treatment goes beyond linear regression by considering the PCA framework. Linear regression is subsumed as a special case, when matrix $\mathbf{U}$ is not necessarily tall but *assumed known*, while $\mathbf{s}_n = \mathbf{s}$, $n = 1, \ldots, N$.

Remark 1: In computer vision applications where robust PCA schemes are particularly attractive, one may not want to discard the entire (vectorized) images $\mathbf{x}_n$, but only specific pixels deemed as outliers [3]. This can be accomplished by replacing $\|\mathbf{O}\|_{2,r}$ in (5) with $\|\mathbf{O}\|_1 := \sum_{n=1}^N \|\mathbf{o}_n\|_1$, a Lasso-type regularization that encourages entry-wise sparsity in $\hat{\mathbf{O}}$.

A. Solving the relaxed problem

To optimize (5) iteratively, an alternating minimization (AM) algorithm is adopted which cyclically updates $\mathbf{S}(k) \rightarrow \mathbf{U}(k) \rightarrow \mathbf{O}(k) \rightarrow \mathbf{m}(k)$ per iteration $k = 1, 2, \ldots$. To update each of the variable groups, (5) is minimized while fixing the rest of the variables to their most up-to-date values.

To derive the updates at iteration k, first form the centered and outlier compensated data matrix $\mathbf{X}_o(k) := \mathbf{X} - \mathbf{1}_N \mathbf{m}(k-1)' - \mathbf{O}(k-1)$. The principal components are readily given by $\mathbf{S}(k) = \mathbf{X}_o(k)\mathbf{U}(k-1)$. Continuing the cycle, $\mathbf{U}(k)$ solves

$$\min_{\mathbf{U}} \|\mathbf{X}_o(k) - \mathbf{S}(k)\mathbf{U}'\|_F^2, \quad \text{s.t. } \mathbf{U}'\mathbf{U} = \mathbf{I}_q$$

a reduced-rank *Procrustes rotation* [19]. The minimizer is given in analytical form in terms of the singular vectors of $\mathbf{X}_o'(k)\mathbf{S}(k)$ [19, Thm. 4]; details under Algortihm 1. Next, the minimization of (5) with respect to $\mathbf{O}$ decouples across rows $\mathbf{o}_n$, resulting in N orthonormal group Lasso problems with respective solutions: $(\mathbf{r}_n(k) := \mathbf{x}_n - \mathbf{m}(k-1) - \mathbf{U}(k)\mathbf{s}_n(k))$

$$\mathbf{o}_n(k) = \mathbf{r}_n(k)(\|\mathbf{r}_n(k)\|_2 - \lambda_2/2)_+ / \|\mathbf{r}_n(k)\|_2, \; n = 1, \ldots, N$$

where $(\cdot)_+ := \max(\cdot, 0)$. These N parallel vector soft-thresholded updates are denoted as $\mathcal{S}(\cdot)$ under Algorithm 1. Finally, the mean update is $\mathbf{m}(k) = (\mathbf{X} - \mathbf{O}(k))'\mathbf{1}_N/N$.

The entire AM solver is tabulated under Algorithm 1, indicating also the recommended initialization. Numerical experiments have shown that few (five to ten) iterations suffice to attain convergence. Algorithm 1 is also conceptually interesting, since it explicitly reveals the intertwining between the outlier identification process, and the PCA low-rank model fitting based on the appropriate outlier compensated data.

Because each of the optimization problems comprising the per iteration cycles has a unique minimizer, and the nondifferentiable regularization only affects one of the variable groups ($\mathbf{O}$); the general convergence results for block-coordinate descent schemes are applicable to Algorithm 1.

Proposition 2: *As $k \rightarrow \infty$ the sequence of iterates generated by Algorithm 1 converges to a stationary point of* (5).

B. Selection of λ_2: robustification paths

Selecting λ_2 controls the number of outliers rejected. But this choice is challenging because existing techniques such as cross-validation are not effective when outliers are present [12]. To this end, systematic approaches can be devised which require either a rough estimate of the percentage of outliers, or, robust estimates $\hat{\sigma}_e^2$ of the nominal noise variance that can be obtained using median absolute deviation schemes [6]. These approaches detailed in [11] leverage the *robustification paths* of (group-)Lasso solutions available for all values of λ_2 [5] to select the one dictated by the data.

Algorithm 1 : Batch robust PCA solver

```
Set $\mathbf{U}(0) = \mathbf{I}_p(:, 1:q)$, $\mathbf{m}(0) = \text{med}(\{\mathbf{x}_n\}_{n=1}^N)$, $\mathbf{O}(0) = \mathbf{0}_{N \times p}$.
for $k = 1, 2, \ldots$ do
    Form $\mathbf{X}_o(k) = \mathbf{X} - \mathbf{1}_N \mathbf{m}'(k-1) - \mathbf{O}(k-1)$.
    Update $\mathbf{S}(k) = \mathbf{X}_o(k)\mathbf{U}(k-1)$.
    Obtain $\mathbf{L}(k)\mathbf{D}(k)\mathbf{R}(k)' = \text{svd}[\mathbf{X}_o'(k)\mathbf{S}(k)]$.
    Update $\mathbf{U}(k) = \mathbf{L}(k)\mathbf{R}'(k)$.
    Update $\mathbf{O}(k) = \mathcal{S}(\mathbf{X} - \mathbf{1}_N \mathbf{m}'(k-1) - \mathbf{S}(k)\mathbf{U}'(k), \lambda_2/2)$.
    Update $\mathbf{m}(k) = (\mathbf{X} - \mathbf{O}(k))'\mathbf{1}_N/N$
end for
```

C. Estimator refinements

Nonconvex regularization. Instead of substituting $\|\mathbf{O}\|_0$ in (4) by its closest convex approximation, namely $\|\mathbf{O}\|_{2,r}$, letting the surrogate function to be nonconvex can yield tighter approximations. To this end, consider approximating (4) by the *nonconvex* formulation

$$\min_{\mathcal{U}, \mathbf{O}} \|\mathbf{X} - \mathbf{1}_N \mathbf{m}' - \mathbf{S}\mathbf{U}' - \mathbf{O}\|_F^2 + \lambda_0 \sum_{n=1}^N \log(\|\mathbf{o}_n\|_2 + \delta)$$
$$\text{s.t. } \mathbf{U}'\mathbf{U} = \mathbf{I}_q. \tag{8}$$

where $\delta \approx 0$ is introduced to avoid numerical instability.

Local methods based on iterative linearization of $\log(\|\mathbf{o}_n\|_2 + \delta)$ around the current iterate $\mathbf{o}_n(k)$, can be adopted to minimize (8) [8]. Skipping details that can be found in [11], this procedure leads to a modified version of Algorithm 1, whereby $\lambda_2 \leftarrow \lambda_0 w_n(k)$ is used for updating each of the $\mathbf{o}_n(k)$. The weights $w_n(k)$ are given by

$$w_n(k) = (\|\mathbf{o}_n(k-1)\|_2 + \delta)^{-1}, \quad n = 1, \ldots, N \tag{9}$$

which altogether amounts to an iteratively reweighted version of (5). To avoid getting trapped in local minima, a good initialization for the iteration is the solution of (5). Extensive numerical tests have shown that even a single iteration of this second stage refinement suffices to yield improved estimates $\hat{\mathcal{U}}$, in comparison to those obtained from (5). The improvements can be leveraged to bias reduction, also achieved by similar *weighted* norm regularizers proposed for linear regression [18].

Outlier rejection. From the equivalence between problems (5) and (6), it follows that those data points $\mathbf{x}_n$ identified as outliers ($\hat{\mathbf{o}}_n \neq \mathbf{0}_p$) are not completely discarded from the estimation process. Instead, their effect is downweighted as per Huber's loss function [cf. (7)]. Nevertheless, explicitly accounting for the outliers in $\hat{\mathbf{O}}$ provides the means of identifying and removing the contaminated data altogether, and thus possibly re-running PCA using the outlier-free data.

IV. Robust Subspace Tracking

Online retailing sites, the World Wide Web, and video surveillance systems generate huge volumes of data, which far outweigh the ability of modern computers to analyze them

Algorithm 2 : Online robust PCA solver

```
Initialize $\mathbf{U}(0) = \mathbf{I}_p(:, 1:q)$ and $\mathbf{s}(0) = \mathbf{0}_q$.
for $n = 1, 2, \ldots$ do
    Update $\mathbf{o}(n) = \mathcal{S}\left(\mathbf{x}_n - \mathbf{U}(n-1)\mathbf{s}(n-1), \lambda_2/2\right)$.
    Update $\mathbf{s}(n) = \mathbf{U}'(n-1)[\mathbf{x}_n - \mathbf{o}(n)]$.
    Update $\mathbf{g}(n) = \mathbf{P}(n-1)\mathbf{s}(n)/[\beta + \mathbf{s}'(n)\mathbf{P}(n-1)\mathbf{s}(n)]$.
    Update $\mathbf{P}(n) = (1/\beta)[\mathbf{P}(n-1) - \mathbf{g}(n)(\mathbf{P}(n-1)\mathbf{s}(n))']$.
    Update $\mathbf{U}(n) = \mathbf{U}(n-1) + [\mathbf{x}_n - \mathbf{U}(n-1)\mathbf{s}(n) - \mathbf{o}(n)]\mathbf{g}'(n)$.
end for
```

in real time. Furthermore, data are generated incrementally in time, which motivates updating previously obtained learning results rather than re-computing new ones from scratch each time a new datum becomes available. This calls for low-complexity real-time (adaptive) algorithms for robust subspace tracking. One possible adaptive counterpart to (5) is the exponentially-weighted LS (EWLS) estimator found by

$$\min_{\{\mathcal{U},\mathbf{O}\}} \sum_{n=1}^{N} \beta^{N-n}\left[\|\mathbf{x}_n - \mathbf{m} - \mathbf{U}\mathbf{s}_n - \mathbf{o}_n\|_2^2 + \lambda_2\|\mathbf{o}_n\|_2\right] \quad (10)$$

where $\beta \in (0, 1]$ is a forgetting factor. Note that in forming the EWLS estimator (10) at time N, the entire history of data $\{\mathbf{x}_n\}_{n=1}^N$ is incorporated in the online estimation process. Whenever $\beta < 1$, past data are exponentially discarded thus enabling operation in nonstationary environments.

Towards deriving a real-time, computationally efficient, and recursive (approximate) solver of (10), an AM scheme will be adopted in which iterations k coincide with the time scale $n = 1, 2, \ldots$ of data acquisition. Per time instant n, a new datum $\mathbf{x}_n$ is drawn and the corresponding $\mathbf{o}(n)$ is updated via soft-thresholding of the residual $\mathbf{r}(n) := \mathbf{x}_n - \mathbf{m}(n-1) - \mathbf{U}(n-1)\mathbf{s}(n-1)$. Only $\mathbf{o}(n)$ is updated at time n, rather than the whole (growing with time) matrix $\mathbf{O}$ that minimization of (10) would dictate. A similar approximate sparse coding step was adopted for online dictionary learning in [10]. Next, the principal component update is $\mathbf{s}(n) = \mathbf{U}'(n-1)[\mathbf{x}_n - \mathbf{m}(n-1) - \mathbf{o}(n)]$, and resembles the projection approximation adopted in [16]. The subspace update is given by

$$\mathbf{U}(n) = \arg\min_{\mathbf{U}} \sum_{i=1}^{n} \beta^{n-i}\|\mathbf{x}_i - \mathbf{m}(i-1) - \mathbf{U}\mathbf{s}(i) - \mathbf{o}(i)\|_2^2$$

and can be efficiently obtained from $\mathbf{U}(n-1)$, via a recursive LS update that capitalizes on the matrix inversion lemma [11]. Note that the orthonormality constraint on the columns of $\mathbf{U}$ is not enforced here, yet the deviation from orthonormality is typically small as observed in [16]. Still, if orthonormal principal directions are required, then an extra orthonormalization step can be carried out per iteration, or, once at the end of the whole process. Finally, $\mathbf{m}(n)$ is obtained recursively as the exponentially-weighted average of the outlier compensated data $\{\mathbf{x}_i - \mathbf{o}(i)\}_{i=1}^n$. The online robust PCA algorithm and its initialization are summarized under Algorithm 2, where $\mathbf{m}$ and its update have been omitted for notational simplicity.

Convergence analysis of Algorithm 2 is beyond the scope of the present paper, and is only supported based on simulations.

Algorithm 3 : Robust KPCA solver

```
Initialize $\mathbf{\Omega}(0) = \mathbf{0}_{N\times N}$ and form $\mathbf{K} = \mathbf{\Phi}'\mathbf{\Phi}$.
for $k = 1, 2, \ldots$ do
    Update $\boldsymbol{\mu}(k) = [\mathbf{I}_N - \mathbf{\Omega}(k-1)]\mathbf{1}_N/N$.
    Form $\mathbf{\Phi}_\Omega(k) = \mathbf{I}_N - \boldsymbol{\mu}(k)\mathbf{1}_N' - \mathbf{\Omega}(k-1)$.
    Form $\tilde{\mathbf{K}}(k) = \mathbf{\Phi}_\Omega'(k)\mathbf{K}\mathbf{\Phi}_\Omega(k)$.
    Update $\mathbf{\Upsilon}(k)$ as the $q$-dominant eigenvectors of $\tilde{\mathbf{K}}(k)$.
    Update $\mathbf{\Sigma}(k) = \mathbf{\Upsilon}'(k)\tilde{\mathbf{K}}(k)$.
    Update $\mathbf{\Omega}(k) = \mathcal{S}\left(\mathbf{I}_N - \boldsymbol{\mu}(k)\mathbf{1}_N' - \mathbf{\Phi}_\Omega(k)\mathbf{\Upsilon}(k)\mathbf{\Sigma}(k), \lambda_2/2\right)$.
end for
```

The numerical tests in Section VI also show that in the presence of outliers, the novel adaptive algorithm outperforms existing nonrobust alternatives for subspace tracking.

V. Robustifying Kernel PCA

Kernel (K)PCA is a generalization to (linear) PCA, seeking principal components in a *feature space* nonlinearly related to the *input space* where the data in $\mathcal{T}$ live [13]. KPCA has been shown effective in performing nonlinear feature extraction for pattern recognition [13]. In addition, connections between KPCA and spectral clustering [5] motivate well the novel KPCA method developed in this section, to robustly identify cohesive subgroups (communities) from social network data.

Consider a nonlinear function $\boldsymbol{\phi} : \mathbf{R}^p \to \mathcal{H}$, that maps elements from the input space $\mathbf{R}^p$ to a feature space $\mathcal{H}$ of arbitrarily large – possibly infinite – dimensionality. The proposed approach to robust KPCA fits the model

$$\boldsymbol{\phi}(\mathbf{x}_n) = \mathbf{m} + \mathbf{U}\mathbf{s}_n + \mathbf{o}_n + \mathbf{e}_n, \quad n = 1, \ldots, N \quad (11)$$

by minimizing (5), given transformed data $\mathcal{T}_\mathcal{H} = \{\boldsymbol{\phi}(\mathbf{x}_n)\}_{n=1}^N$.

Except for $\mathbf{S}$, the data as well as the unknowns in (5) are now vectors/matrices of infinite dimension. In principle, this challenges the optimization task since it is not possible to store, or, perform updates of such quantities directly. This hurdle can be overcome by endowing $\mathcal{H}$ with the structure of a reproducing kernel Hilbert space (RKHS), where inner products between any two members of $\mathcal{H}$ boil down to evaluations of the reproducing kernel $K_\mathcal{H} : \mathbf{R}^p \times \mathbf{R}^p \to \mathbb{R}$, i.e., $\langle\boldsymbol{\phi}(\mathbf{x}_i), \boldsymbol{\phi}(\mathbf{x}_j)\rangle_\mathcal{H} = K(\mathbf{x}_i, \mathbf{x}_j)$. This so-termed *kernel trick* is the crux of most kernel methods in machine learning [5], including kernel PCA [13]. The problem of selecting a suitable kernel $K_\mathcal{H}$ (and $\boldsymbol{\phi}$ indirectly) will not be considered here.

Building on these ideas, a main result in [11] states that the iterations of a provably convergent AM solver of (5) cycling through $\mathbf{m}(k) \to \mathbf{U}(k) \to \mathbf{S}(k) \to \mathbf{O}(k)$, can be equivalently carried out in terms of *finite-dimensional* 'sufficient statistics' $\boldsymbol{\mu}(k) \to \mathbf{\Upsilon}(k) \to \mathbf{\Sigma}(k) \to \mathbf{\Omega}(k)$. The latter updates are tabulated as Algorithm 3, where $\mathbf{K} := \mathbf{\Phi}'\mathbf{\Phi} \in \mathbb{R}^{N\times N}$ is the kernel matrix to be computed offline, and $\mathbf{\Phi} := [\boldsymbol{\phi}(\mathbf{x}_1), \ldots, \boldsymbol{\phi}(\mathbf{x}_N)]$. Because $\mathbf{O}(k) = \mathbf{\Phi}\mathbf{\Omega}(k)$ [11], the outlier vector norms required to perform outlier sparsity control are computable in terms of $\mathbf{K}$, i.e., $[\|\mathbf{o}_1(\infty)\|_2^2, \ldots, \|\mathbf{o}_N(\infty)\|_2^2]' = \text{diag}[\mathbf{\Omega}'(\infty)\mathbf{K}\mathbf{\Omega}(\infty)]$. Moreover, for any given new data point $\mathbf{x}$, its principal component

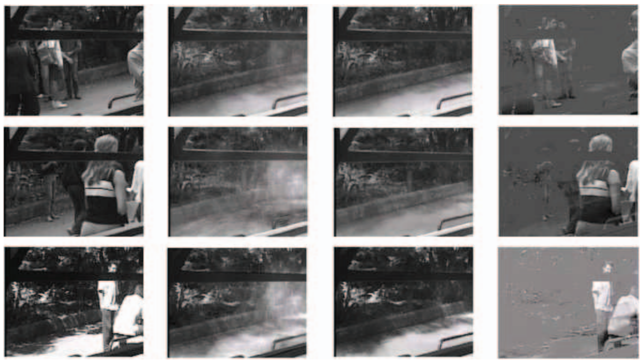

Fig. 1. Background modeling for video surveillance. First column: original frame $\mathbf{x}_n$; Second column: PCA reconstruction; Third column: robust PCA reconstruction; Fourth column: outliers in $\hat{\mathbf{o}}_n$.

in feature space is given by $\mathbf{s} = \boldsymbol{\Upsilon}'(\infty)\boldsymbol{\Phi}'_{\Omega}(\infty)\boldsymbol{\Phi}'\phi(\mathbf{x})$, which is again computable in terms of the kernel function $K_{\mathcal{H}}$.

VI. Numerical Tests

Video surveillance. To validate the proposed approach to robust PCA, Algorithm 1 was tested to perform background modeling from a sequence of video frames; an approach that has found widespread applicability for intrusion detection in video surveillance systems. The experiments were carried out using the dataset studied in [3], which consists of $N = 520$ images ($p = 120 \times 160$) acquired from a static camera during two days. The illumination changes considerably over the two day span, while approximately 40% of the training images contain people in various locations. For $q = 10$, both standard PCA and the robust PCA of Section III were applied to build a low-rank background model of the environment captured by the camera. For robust PCA, ℓ_1-norm regularization on $\mathbf{O}$ was adopted to identify outliers at a pixel level. The outlier sparsity-controlling parameter was chosen as $\lambda_2 = 9.69 \times 10^{-4}$, whereas a single iteration of the reweighted scheme in Section III-C was run to the reduce the bias in $\hat{\mathbf{O}}$.

The results are shown in Fig. 1, for three representative images. The first column comprises the original frames from the training set, while the second column shows the corresponding (nonrobust) PCA image reconstructions. The presence of undesirable 'ghostly' artifacts is apparent, since PCA is not able to completely separate the people from the background. The third column illustrates the robust PCA reconstructions, which recover the illumination changes while successfully subtracting the people. The fourth column shows the reshaped outlier vectors $\hat{\mathbf{o}}_n$, which mostly capture the people and abrupt changes in illumination.

Robust subspace tracking. A simulated test is carried out here to corroborate the convergence and effectiveness of the robust online PCA algorithm in Section IV. For $N = 1000$, $p = 100$, and $q = 30$, nominal data in $\mathcal{T}$ are generated according to model (1), where $\mathbf{e}_n \sim \mathcal{N}(\mathbf{0}_p, 10^{-3}\mathbf{I}_p)$. The first ten entries of $\mathbf{x}_{201}, \ldots, \mathbf{x}_{205}$ are outliers, uniformly distributed in $[-100, 100]$ and i.i.d.. Fig. 2 depicts the evolution of the angle formed between the learnt subspace (spanned by the columns of) $\mathbf{U}(n)$ and the true subspace $\mathbf{U}$ generating $\mathcal{T}$.

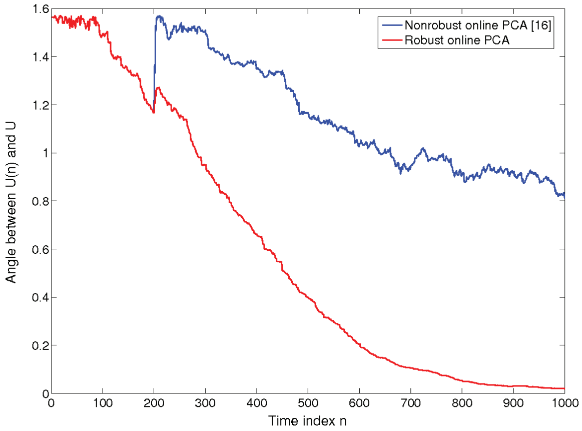

Fig. 2. Time evolution of the angle between the learnt subspace $\mathbf{U}(n)$, and the true $\mathbf{U}$ used to generate the data ($\beta = 0.999$ and $\lambda_2 = 3.3$).

Convergence of Algorithm 2 to $\mathbf{U}$ is apparent, and it markedly outperforms the nonrobust subspace tracking method in [16].

References

[1] N. A. Campbell, "Robust procedures in multivariate analysis I: Robust covariance estimation," *Applied Stat.,* vol. 29, pp. 231-237, 1980.

[2] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," 2009 (submitted).

[3] F. de la Torre and M. J. Black, "A framework for robust subspace learning," *Int. Jrnl. of Computer Vision,* vol. 54, pp. 183-209, 2003.

[4] J. J. Fuchs, "An inverse problem approach to robust regression," in *Proc. of ICASSP*, Phoeniz, AZ, Mar. 1999.

[5] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York, NY: Springer, 2009.

[6] P. J. Huber and E. Ronchetti, *Robust Statistics,* New York: Wiley, 2009.

[7] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York, NY: Springer, 2002.

[8] V. Kekatos and G. B. Giannakis, "Robust layered sensing: From sparse signals to sparse residuals," in *Proc. of 44th Asilomar Conf.*, Pacific Grove, CA, Nov. 2010; http://arxiv.org/abs/1011.0450

[9] H. Kim, S. Lee, X. Ma, and C. Wang, "Higher-order PCA for anomaly detection in large-scale networks," in *Proc. of CAMSAP*, Aruba, Dutch Antilles, Dec. 2009.

[10] J. Mairal, J. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Jrnl. of Machine Learning Research*, vol. 11, pp. 19-60, Jan. 2010.

[11] G. Mateos and G. B. Giannakis, "Sparsity control for robust principal component analysis," *IEEE Trans. Sig. Proc.*, 2010 (submitted).

[12] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, New York, NY: Wiley, 1987.

[13] B. Schlkopf, A. Smola, and K.-R. Mller, "Kernel principal component analysis," in *Artificial Neural Networks: Lec. Notes in Computer Science,* vol. 1327, pp. 583-588, Berlin: Springer, 1997.

[14] J. Tropp, "Just relax: Convex programming methods for identifying sparse signals," *IEEE Trans. Info. Theory*, vol. 52, no. 3, pp. 1030-1051, Mar. 2006.

[15] L. Xu and A. L. Yuille, "Robust principal component analysis by self-organizing rules based on statistical physics approach," *IEEE Trans. Neural Nets.*, vol. 6, no. 1, pp. 131-143, Jan. 1995.

[16] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Sig. Proc.*, vol. 43, no. 1, pp. 95-107, Jan. 1995.

[17] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Jrnl. of the Royal Stat. Society, Series B*, vol. 68, no. 1, pp. 4967, 2006.

[18] H. Zou, "The adaptive Lasso and its oracle properties," *Jrnl. of the American Stat. Assoc.*, vol. 101, no. 476, pp. 1418-1429, Dec. 2006.

[19] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Jrnl. of Comp. and Graphical Statistics*, vol. 15, no. 2, pp. 265-286, 2006.