Novel Algorithm for Incremental L1-Norm Principal-Component Analysis

Mayur Dhanaraj and Panos P. Markopoulos
Dept. of Electrical and Microelectronic Engineering, Rochester Institute of Technology, Rochester, NY 14623, USA
E-mails: {mxd6023, panos}@rit.edu

***Abstract*—L1-norm Principal-Component Analysis (L1-PCA) has been shown to exhibit sturdy resistance against outliers among the processed data. In this work, we propose L1-IPCA: an algorithm for incremental L1-PCA, appropriate for big-data and streaming-data applications. The proposed algorithm updates the calculated L1-norm principal components as new data points arrive, conducting a sequence of computationally efficient bit-flipping iterations. Our experimental studies on subspace estimation, image conditioning, and video foreground extraction illustrate that the proposed algorithm attains remarkable outlier resistance at low computational cost.**

***Index Terms*—Image/video processing, incremental PCA, L1-norm PCA, outliers, online learning.**

I. Introduction

Principal-component analysis (PCA) [1] is a data analysis method with numerous applications in image/video processing, computer vision, machine learning, and pattern recognition, among many other fields [2], [3]. Broadly, PCA seeks a number of orthogonal directions, known as principal components (PCs), that define a subspace wherein data presence is maximized. In its standard form, PCA quantifies data presence by the squared L2-norm (sum of squared entries) of the subspace-projected data points. Mathematically, given data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ with rank $d \leq \min\{D, N\}$ and a number of PCs $K \leq d$, PCA is formulated as [4]

$$\mathbf{Q}_{L2} = \underset{\substack{\mathbf{Q} \in \mathbb{R}^{D \times K} \\ \mathbf{Q}^T \mathbf{Q} = \mathbf{I}_K}}{\operatorname{argmax}} \sum_{i=1}^{N} \|\mathbf{Q}^T \mathbf{x}_i\|_2^2, \tag{1}$$

The great popularity of PCA can be partially attributed to its low-cost and widely accessible solution by means of singular-value decomposition (SVD) [5].

On the negative side, it is long observed that PCA is very sensitive against faulty data points that lie far away from the nominal measurements, known as outliers [6]. This sensitivity of PCA is due to its reliance on the L2-norm, which places quadratic emphasis on the magnitude of each data point, benefiting peripheral/outlying ones. Regretfully, outliers are encountered in many real-world big-data applications due to various causes, such as sensor malfunctions, sporadic interference, and errors in data transcription and labeling. To counteract outliers, an L1-norm-based reformulation of standard PCA has been proposed. Replacing the L2-norm in (1) by the L1-norm, L1-PCA is formulated as [7]

$$\mathbf{Q}_{L1} = \underset{\substack{\mathbf{Q} \in \mathbb{R}^{D \times K} \\ \mathbf{Q}^T \mathbf{Q} = \mathbf{I}_K}}{\operatorname{argmax}} \sum_{i=1}^{N} \|\mathbf{Q}^T \mathbf{x}_i\|_1, \tag{2}$$

where the L1-norm (sum of the absolute entries) $\|\cdot\|_1$ is less responsive to the high-valued entries of its argument. So far, L1-PCA has demonstrated high outlier resistance in many signal processing and machine learning applications. Early approximate solvers for (2), with cost similar to SVD, were presented in [8]–[10]. In [7], authors offered the first exact solution to L1-PCA in (2), with cost $\mathcal{O}(N^{dK-K+1})$ (for d constant with respect to N). Most recently, authors in [11] introduced a state-of-the-art approximate algorithm for L1-PCA with cost $\mathcal{O}(ND\min\{N, D\} + N^2(K^4 + DK^2) + NDK^3)$ –comparable to the cost of SVD for solving PCA in (1). Importantly, the algorithm of [11] attains small (if any) performance degradation in the metric of L1-PCA, outperforming many preexisting low-cost counterparts.

At the same time, it is understood that many real-world applications call for *incremental* PCA solutions. This is, e.g., when data-size N in (1) is too large for efficient *batch* processing (big-data applications), or when the entire dataset is unavailable initially and data points (columns of $\mathbf{X}$) arrive in a streaming fashion (*e.g.,* in online video processing [12]–[14], or dynamic face-ID [15]). Clearly, in the case of data streaming, it would be computationally inefficient to append the new data points next to the previously collected ones and update the PCs with a new PCA/SVD on the augmented batch from scratch.

To analyze big and/or streaming data, an array of incremental PCA algorithms have been proposed in literature [16]–[18]. The majority of these algorithms perform well when the processed data are corruption-free. However, relying to the L2-norm PCA formulation of (1), most of these algorithms tend to be sensitive against outliers amongst the processed data. Thus, the need rises for incremental PCA calculators that also offer sturdy outlier resistance. To this end, the first incremental L1-PCA solvers in the literature were proposed in [19], [20], in the context of video foreground extraction, and [21], in the context of visual object tracking. Other works propose incremental robust-PCA that considers the processed matrix to be summation of a low-rank component of interest and a sparse component that models

Work in this paper was supported by the US AFOSR, Dynamic Data Driven Applications Systems (DDDAS) program, grant FA9550-18-1-0121.

ISBN 978-90-827970-1-5 © EURASIP 2018

L1-BF Algorithm

Input: $\mathbf{X} \in \mathbb{R}^{D \times N}$, $\mathbf{B}^{\text{init}} \in \{\pm 1\}^{N \times K}$ and $K \leq d = \text{rank}(\mathbf{X})$

$\mathbf{B}_{\text{bf}} \leftarrow \texttt{BF}(\mathbf{X}, \mathbf{B}^{\text{init}}, K)$
$(\mathbf{U}, \mathbf{\Sigma}_{K \times K}, \mathbf{V}) \leftarrow \texttt{SVD}(\mathbf{X}\mathbf{B}_{\text{bf}})$
$\mathbf{Q}_{\text{bf}} \leftarrow \mathbf{U}\mathbf{V}^T$

Output: $\mathbf{Q}_{\text{bf}}$

Function: $\mathbf{B} \leftarrow \texttt{BF}(\mathbf{X}_{D \times N}, \mathbf{B}_{N \times K}, K)$

$\omega \leftarrow K||\mathbf{X}[\mathbf{B}]_{:,1}||_2$
while true (or terminate at NK iterations)
  for $m \in \{1, 2, \ldots, N\}$, $l \in \{1, 2, \ldots, K\}$
    $a_{m,l} \leftarrow ||\mathbf{X}\mathbf{B} - 2\mathbf{B}_{m,l}\mathbf{x}_m\mathbf{e}_{l,k}^T||_*$
  $(n, p) \leftarrow \text{argmax}_{m,l}\ a_{m,l}$
  if $\omega < a_{n,p}$
    $\mathbf{B}_{n,p} \leftarrow -\mathbf{B}_{n,p}$, $\omega \leftarrow a_{n,p}$
  else, break
Return $\mathbf{B}$

Fig. 1: Pseudo-code of L1-BF algorithm [11] for the approximate computation of K L1-PCs of $\mathbf{X}$ (MATLAB code available in [25]).

the outliers [12], [13], [22], [23]. Robust Grassman-manifold tracking algorihtms [24] have also been proposed.

Inspired by the algorithms of [19]–[21], in this work we propose L1-IPCA: an algorithm for incremental L1-PCA that attains remarkable outlier resistance, high convergence speed, and low computational cost. Our experimental studies on subspace estimation, video processing, and image conditioning illustrate the superior performance and efficiency of the proposed algorithm, compared with popular alternatives.

II. Brief Review of Exact and Approximate Batch L1-PCA

A. Exact L1-PCA

In [7] it was shown that if

$$\mathbf{B}_{\text{opt}} = \underset{\mathbf{B} \in \{\pm 1\}^{N \times K}}{\text{argmax}} \ ||\mathbf{X}\mathbf{B}||_*, \tag{3}$$

where nuclear norm $||.||_*$ returns the sum of the singular values of its matrix argument, and $\mathbf{X}\mathbf{B}_{\text{opt}} \overset{\text{SVD}}{=} \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, then $\mathbf{Q}_{\text{L1}} = \mathbf{U}\mathbf{V}^T$ is the optimal solution to the L1-PCA problem in (2). That is, L1-PCA can be equivalently reformulated as a combinatorial optimization problem, searching for an optimal antipodal binary matrix $\mathbf{B}_{\text{opt}} \in \{\pm 1\}^{N \times K}$ that solves (3). In [7] it was shown that the cost for solving (3), and therefore L1-PCA in (2), is $\mathcal{O}(2^{NK})$ in general and $\mathcal{O}(N^{dK-K+1})$ when d is a constant with respect to N.

B. L1-PCA via Bit-flipping

In order to compute the K L1-PCs of $\mathbf{X} \in \mathbb{R}^{D \times N}$ with reduced computational cost, authors in [11] proposed L1-BF: a suboptimal algorithm that approximates $\mathbf{B}_{\text{opt}} \in \{\pm 1\}^{N \times K}$ by an antipodal binary matrix $\mathbf{B}_{\text{bf}} \in \{\pm 1\}^{N \times K}$ obtained through a sequence of bit-flipping (BF) iterations. Specifically, L1-BF initializes at an arbitrary $\mathbf{B}^{\text{init}} \in \{\pm 1\}^{N \times K}$ (or smarter initialization for faster convergence) and executes a sequence of BF iterations, such that the metric in (3) increases monotonically. That is, at the t-th iteration, L1-BF browses all bits that are not flipped since initialization and identifies the single bit that, when flipped, offers the highest increase in the metric of (3) as

$$(n, p) = \underset{(m,l) \in \{1,2,\ldots,N\} \times \{1,2,\ldots,K\}}{\text{argmax}} ||\mathbf{X}\mathbf{B}^{(t)} - 2\mathbf{B}_{m,l}^{(t)}\mathbf{x}_m\mathbf{e}_{l,K}^T||_*, \tag{4}$$

where $\mathbf{x}_m$ is the m-th column of data matrix X and $\mathbf{e}_{l,K}$ denotes the l-th column of the size-K identity matrix. Then, L1-BF flips (n, p)-th bit by setting $\mathbf{B}^{(t+1)} = \mathbf{B}^{(t)} - 2\mathbf{B}_{n,p}^{(t)}\mathbf{e}_{n,N}\mathbf{e}_{p,K}^T$. Iterations terminate and $\mathbf{B}_{\text{bf}}$ is returned when the metric in (3) cannot further increase by any single bit flip. Finally, $\mathbf{B}_{\text{bf}}$ is used to approximate the K L1-PCs of $\mathbf{X}$ as

$$\mathbf{Q}_{\text{bf}} = \mathbf{U}\mathbf{V}^T, \tag{5}$$

where $\mathbf{X}\mathbf{B}_{\text{bf}} \overset{\text{SVD}}{=} \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. L1-BF constitutes an important building block of the algorithm proposed in this work. Therefore, we offer a pseudocode for L1-BF in Fig. 1.

III. Proposed Algorithm for Incremental L1-PCA

The proposed algorithm strives to approximate incrementally the outlier-free PCs of a given data matrix $\mathbf{X} \in \mathbb{R}^{D \times N}$. We commence by collecting few data points in the small batch matrix $\mathbf{X}_B^{(0)} \in \mathbb{R}^{D \times N_B}$ with $\text{rank}(\mathbf{X}_B^{(0)}) \geq K$. Then, we compute/approximate the K L1-PCs of $\mathbf{X}_B^{(0)}$, $\mathbf{Q}_{\text{bf}}^{(0)} \in \mathbb{R}^{D \times K}$, by means of the L1-BF procedure of Fig. 1. Subsequently, all other columns of $\mathbf{X}$ are processed one-by-one in a streaming fashion.

When the new point $\mathbf{x}_{\text{in}}^{(t)} = [\mathbf{X}]_{:,N_B+t}$, $t = 1, 2, \ldots, N - N_B$ arrives for processing, it is first subjected to an L1-PCA-informed reliability evaluation [26]–[28]. Specifically, the reliability of $\mathbf{x}_{\text{in}}^{(t)}$ is quantified as the normalized magnitude of its projection onto the already available L1-PCs $\mathbf{Q}_{\text{bf}}^{(t-1)} \in \mathbb{R}^{D \times K}$,

$$r(\mathbf{x}_{\text{in}}^{(t)}) = \frac{||\mathbf{Q}_{\text{bf}}^{(t-1)^T}\mathbf{x}_{\text{in}}^{(t)}||_2^2}{||\mathbf{x}_{\text{in}}^{(t)}||_2^2}. \tag{6}$$

We observe that $0 \leq r(\mathbf{x}_{\text{in}}^{(t)}) \leq 1$. Clearly, high reliability, close to 1, indicates that $\mathbf{x}_{\text{in}}^{(t)}$ is coherent with the previously calculated $\mathbf{Q}_{\text{bf}}^{(t-1)}$. Subsequently, the calculated reliability is compared with a predefined threshold τ. If $r(\mathbf{x}_{\text{in}}^{(t)}) \leq \tau$, then $\mathbf{x}_{\text{in}}^{(t)}$ is considered to be a possible outlier with respect to $\mathbf{Q}_{\text{bf}}^{(t-1)}$ and, therefore, it is discarded. If, on the other hand, $r(\mathbf{x}_{\text{in}}^{(t)}) > \tau$, then $\mathbf{x}_{\text{in}}^{(t)}$ is admitted for processing towards updating the L1-PCs in $\mathbf{Q}_{\text{bf}}^{(t-1)}$. Specifically, the updated $\mathbf{Q}_{\text{bf}}^{(t)}$ is set to the L1-PCs of the augmented batch

$$\mathbf{Y}^{(t)} = [\mathbf{X}_B^{(t-1)}, \ \mathbf{x}_{\text{in}}^{(t)}] \in \mathbb{R}^{D \times (N_B+1)}. \tag{7}$$

Instead of calculating the L1-PCs of $\mathbf{Y}^{(t)}$ from scratch, the proposed algorithm incrementally adapts the previously computed L1-PCs in $\mathbf{Q}_{\text{bf}}^{(t-1)}$, towards both complexity savings and performance enhancement. Specifically, in this adaptation step, we first calculate

$$\mathbf{B}^{\text{init}^{(t)}} = \text{sgn}(\mathbf{Y}^{(t)^T}\mathbf{Q}_{\text{bf}}^{(t-1)}). \tag{8}$$

ISBN 978-90-827970-1-5

Then we run L1-BF iterations on $\mathbf{Y}^{(t)}$, initialized at $\mathbf{B}^{\text{init}^{(t)}}$, as demonstrated in the L1-BF update (L1-BFU) routine of Fig. 2. Upon convergence (or termination –see line 2 of function `BF` in Fig. 1), L1-BFU returns the new, incrementally updated L1-PCs, $\mathbf{Q}_{\text{bf}}^{(t)}$.

After $\mathbf{Q}_{\text{bf}}^{(t)}$ is computed, the reliability of each point in the current augmented batch $\mathbf{Y}^{(t)}$ is re-evaluated as

$$r([\mathbf{Y}^{(t)}]_{:,j}) = \frac{||\mathbf{Q}_{\text{bf}}^{(t)^T}[\mathbf{Y}^{(t)}]_{:,j}||_2^2}{||[\mathbf{Y}^{(t)}]_{:,j}||_2^2}, \tag{9}$$

$j = 1, 2, \ldots, N_B + 1$. To maintain constant memory/storage and computational demands, we remove from the augmented batch $\mathbf{Y}^{(t)}$ the column (data point) that attains the lowest reliability with respect to the updated L1-PCs in $\mathbf{Q}_{\text{bf}}^{(t)}$. That is, we calculate

$$j^* = \underset{j \in \{1,2,\ldots,N_B+1\}}{\text{argmin}} \; r([\mathbf{Y}^{(t)}]_{:,j}) \tag{10}$$

and then update the batch as

$$\mathbf{X}_B^{(t)} = [\mathbf{Y}^{(t)}]_{:,1:j^*-1,j^*+1:N_B+1}. \tag{11}$$

Thus, the size of the carried over batch $\mathbf{X}_B^{(t)}$ is maintained constant across t, equal to $D \times N_B$.

The proposed algorithm can be viewed as a modified version of the incremental L1-PCA calculator proposed in [19]. Specifically, L1-IPCA differs from the algorithm of [19] in two main ways. First, in contrast to [19], L1-IPCA applies an L1-informed reliability pre-check by means of which it rejects incoming points that appear to be outliers. This feature adds a very effective second layer of defense against faulty/corrupted data points, which leads to superior subspace estimation with fewer data points processed. Second, L1-IPCA attains significantly reduced computational cost, by initializing the BF iterations at the $\mathbf{B}^{\text{init}^{(t)}}$ of (8), instead of the exhaustively derived initialization of [19].

Proposed L1-IPCA

Input: $\mathbf{X} \in \mathbb{R}^{D \times N}$, $K \leq \text{rank}(\mathbf{X})$ and τ

$\mathbf{X}_B = [\mathbf{X}]_{:,1:N_B}$ $\quad$ (N_B such that $\text{rank}(\mathbf{X}_B) \geq K$)
$\mathbf{B}^{\text{init}} \leftarrow$ arbitrary
$\mathbf{Q}_{\text{bf}} \leftarrow$ `L1-BF`$(\mathbf{X}_B, \mathbf{B}^{\text{init}}, K)$
for $t = 1, 2, \ldots, N - N_B$
 $\quad \mathbf{x}_{\text{in}} = [\mathbf{X}]_{:,N_B+t}$
 $\quad (\mathbf{Q}_{\text{bf}}, \mathbf{X}_B) \leftarrow$ `L1-BFU` $(\mathbf{X}_B, \mathbf{x}_{\text{in}}, K, \mathbf{Q}_{\text{bf}}, \tau)$

Output: $\mathbf{Q}_{\text{bf}}$

Function: $(\mathbf{Q}_{\text{bf}}, \mathbf{X}_B) \leftarrow$ `L1-BFU` $(\mathbf{X}_B, \mathbf{x}_{\text{in}}, K, \mathbf{Q}_{\text{bf}}, \tau)$

r_{in} = `reliability`$(\mathbf{x}_{\text{in}}, \mathbf{Q}_{\text{bf}})$
if $r_{in} > \tau$
 $\quad \mathbf{Y} \leftarrow [\mathbf{X}_B, \mathbf{x}_{\text{in}}]$
 $\quad \mathbf{B}^{\text{init}} \leftarrow \text{sgn}(\mathbf{Y}^T \mathbf{Q}_{\text{bf}})$
 $\quad \mathbf{Q}_{\text{bf}} \leftarrow$ `L1-BF`$(\mathbf{Y}, \mathbf{B}^{\text{init}}, K)$
 $\quad \mathbf{r} \leftarrow$ `reliability`$(\mathbf{Y}, \mathbf{Q}_{\text{bf}})$
 $\quad j^* \leftarrow \text{argmin}_{j \in \{1,2,\ldots,N_B+1\}} r_j$
 $\quad \mathbf{X}_B \leftarrow [\mathbf{Y}]_{:,[1:j^*-1,\, j^*+1:N_B+1]}$
Return $\mathbf{Q}_{\text{bf}}$, $\mathbf{X}_B$

Function: $\mathbf{r} \leftarrow$ `reliability` $(\mathbf{A}, \mathbf{Q})$

for $j = 1, 2, \ldots, N =$ `size`$(\mathbf{A}, 2)$
 $\quad \mathbf{a} \leftarrow [\mathbf{A}]_{:,j}$
 $\quad r_j \leftarrow \frac{||\mathbf{Q}^T \mathbf{a}||_2^2}{||\mathbf{a}||_2^2}$
Return $\mathbf{r} = [r_1, r_2, \ldots, r_N]^T$

Fig. 2: Proposed L1-IPCA algorithm for incremental L1-PCA.

IV. Experimental Studies

A. Subspace Estimation

In this first study, we evaluate the performance of the proposed algorithm on synthetic data. We set $D = 5$ and define $\mathbf{s} \in \mathbb{R}^{(D=5) \times 1}$, with $\|\mathbf{s}\|_2 = 1$, and the nominal data distribution $\mathcal{N}(\mathbf{0}_5, \alpha \mathbf{s}\mathbf{s}^T)$, for some $\alpha > 0$. We draw $N = 250$ points from this nominal distribution and form data matrix $\mathbf{X} \in \mathbb{R}^{5 \times 250}$, verifying that $\text{rank}(\mathbf{X}) = D = 5$. Next, we define $\mathbf{z} \in \mathbb{R}^{5 \times 1}$, with $\|\mathbf{z}\|_2 = 1$ and $\arccos(\mathbf{z}^T \mathbf{s}) = 78°$, and the outlier distribution $\mathcal{N}(0_5, \beta \mathbf{z}\mathbf{z}^T)$, with $\frac{\beta}{\alpha} = 30$ (*i.e.,* outliers have 30 times higher variance than nominal points). We replace few of the columns of data matrix $\mathbf{X}$ with points from the outlier distribution. Then, we set $N_B = 20, \tau = 0.55$, and $K = 1$ and run the proposed incremental L1-PCA algorithm, seeking to identify the nominal data basis $\mathbf{s}$. We consider that 1 out of the 20 points in $\mathbf{X}_B^{(0)}$ and 3 out of the remaining 230 points in $\mathbf{X}$ are outliers. The performance of our algorithm is evaluated by means of the standard subspace dissimilarity metric

$$e^{(t)} = 1 - \frac{||\mathbf{s}^T \mathbf{q}^{(t)}||_2^2}{||\mathbf{q}^{(t)}||_2^2} \tag{12}$$

where $\mathbf{q}^{(t)} \in \mathbb{R}^{5 \times 1}$ is the incrementally calculated L1-PC after the t-th update measurement is processed. Ideally, we want $e^{(t)}$ to be as close to zero as possible. We repeat the experiment 1000 times and plot in Fig. 3a the average $e^{(t)}$ versus update measurement index t. Together with the proposed algorithm, we plot the performance (on the exact same data) of ISVD [17], SVD (SVD anew on the entire $[\mathbf{X}]_{:,1:t}$, for every t), the GRASTA algorithm of [24], the method of [18], and the incremental L1-PCA of [21]. We observe that [17], [18], and SVD start at relatively high average dissimilarity due to the presence of one outlier in the first 20 points and the outlier-sensitivity of their L2-norm initialization. Their performance improves as they process nominal points and deteriorates instantly each time they process an outlier (depicted by dotted vertical lines). GRASTA [24] starts at an intermediate point due to its arbitrary initialization and adapts well until an outlier appears, when it also deviates immediately from the nominal subspace. The algorithm of [21] starts from a low dissimilarity point, as it is initialized on outlier-resistant L1-PCs (calculated by means of L1-BF on $\mathbf{X}_B^{(0)}$), and adapts well until an outlier occurs. Then it momentarily deviates and recovers quickly when new

ISBN 978-90-827970-1-5

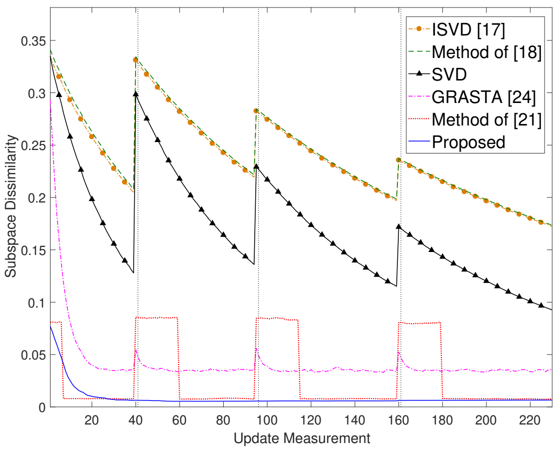

Fig. 3a: Subspace dissimilarity versus update measurement index ($K = 1$, $D = 5$, $N = 250$, and $N_B = 20$).

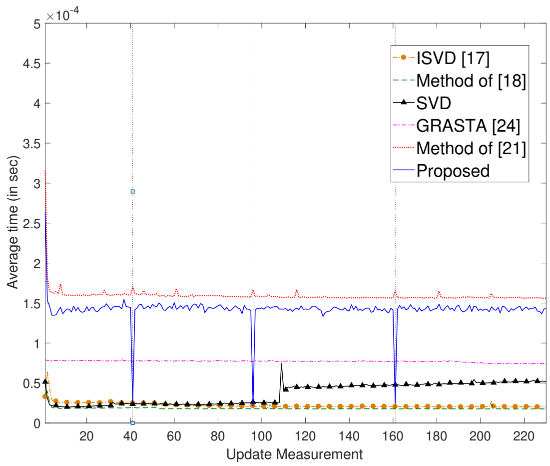

Fig. 3b: Cumulative time versus update measurement for $K = 1$ ($D = 5$, $N = 250$ and $N_B = 20$).

nominal points arrive. The proposed method also starts from a low subspace dissimilarity point and then it stays robust against outliers while it converges fast to the lowest subspace dissimilarity across the board.

In Fig. 3b, we plot the average time needed by each algorithm for PC adaptation, versus the update measurement index t. We observe that all methods display similar average computation time. GRASTA [24], ISVD [17] and [18] display consistently low computation time, while the computation time of batch SVD increases with the number of measurements processed. The proposed algorithm is slightly faster than the method of [21]. Interestingly, the average computation time of L1-IPCA drops significantly at the indices that correspond to the three outliers, as they are rejected by means of the L1-PCA-informed pre-check and they trigger no PC-update processing.

B. Image Conditioning

In this experiment, we perform image conditioning by means of L1-IPCA. Specifically, we first select 13 images of a person's face from the PICS database [29], captured under varying illumination conditions and exhibiting glare and shadow artifacts. Then, we crop all images to 200×200 pixels, vectorize them, and arrange them as columns of data matrix $\mathbf{X} \in \mathbb{R}^{40000 \times 13}$. Our goal is to extract the background image (face) by means of L1-PCA, eliminating glare and shadow artifacts as foreground outliers.

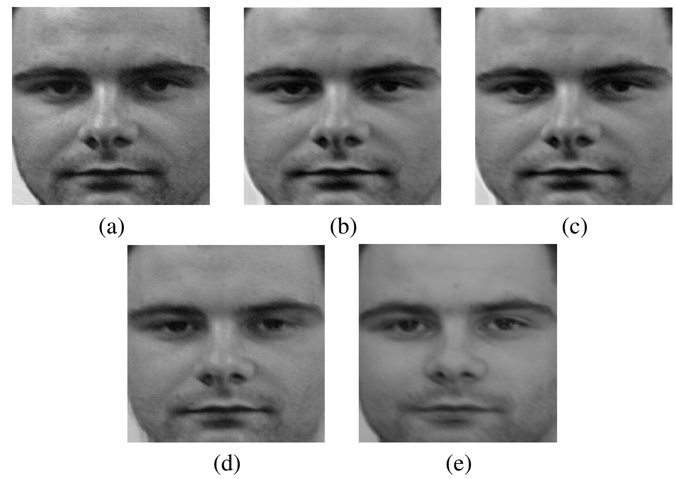

Fig. 4: (a) Original face image with glare. Image conditioned with (b) ISVD [17], (c) the method of [18], (d) GRASTA [24], and (e) the proposed L1-IPCA.

We set $N_B = K = 4$ and $\tau = 0.975$ and apply the L1-IPCA method to obtain, after all 13 images are processed, the L1-PCs $\mathbf{Q}_{\text{bf}}$. To remove undesired illumination artifacts, each vectorized image $\mathbf{x}_i = [\mathbf{X}]_{:,i}$ is projected on $\mathbf{Q}_{\text{bf}}$ as $\mathbf{Q}_{\text{bf}}\mathbf{Q}_{\text{bf}}^T\mathbf{x}_i$. Our image conditioning results are presented in Fig. 4. The proposed method is compared with ISVD [17], GRASTA [24], and the method of [18]. We observe that [17], [18] retains most of the glare and shadows in the original face image. GRASTA [24] performs comparatively better glare rejection. Our algorithm appears to outperform all alternatives, eliminating glare and limiting significantly the shadows.

C. Dynamic Video Foreground Extraction

Video foreground extraction finds important applications in, *e.g.,* security surveillance, human-computer interaction, and traffic monitoring [30], [31]. The static background of each frame comprises the principal subspace whereas moving objects (*e.g.,* people, vehicles) in the foreground constitute numerical outliers. In this experiment, we use a video of $N = 438$ 202×269 frames, included in the popular CAVIAR database [32]. Each video frame is vectorized and arranged as a column of data matrix $\mathbf{X} \in \mathbb{R}^{54338 \times 438}$. We set $N_B = 5$, $K = 5$, and $\tau = 0.985$ and run the proposed algorithm on $\mathbf{X}$ to calculate incrementally its L1-PCs in $\mathbf{Q}_{\text{bf}} \in \mathbb{R}^{D \times K}$. Then, the static video background in the i-th frame $\mathbf{x}_i = [\mathbf{X}]_{:,i}$ is identified by the projection $\mathbf{Q}_{\text{bf}}\mathbf{Q}_{\text{bf}}^T\mathbf{x}_i$. Accordingly, the foreground of the frame is extracted as $\mathbf{x}_i - \mathbf{Q}_{\text{bf}}\mathbf{Q}_{\text{bf}}^T\mathbf{x}_i$. The calculated background and foreground components of the 200-th frame of the processed video are presented in Fig. 5. The results of the proposed L1-IPCA are compared with those of ISVD [17], the method of [18], and GRASTA [24]. The methods of [16]–

ISBN 978-90-827970-1-5 © EURASIP 2018

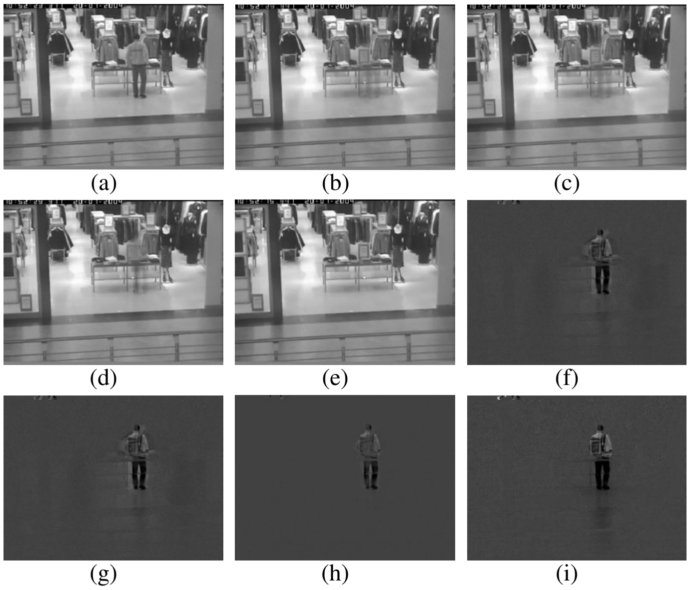

Fig. 5: (a) 200-th frame of the processed video. Background extracted by (b) ISVD [17], (c) method of [18], (d) GRASTA [24], and (e) proposed L1-IPCA. Foreground extracted by (f) ISVD [17], (g) method of [18], (h) GRASTA [24], and (i) proposed L1-IPCA.

[18] exhibit, in different extents, visible ghostly appearances of the man in the background component. However, the proposed algorithm manages to identify successfully the background and, accordingly, extract a significantly more precise foreground (both the man and his shadow, with accurately fitted outline), compared to its counterparts.

V. Conclusions

We presented L1-IPCA: an algorithm for incremental L1-PCA, appropriate for big-data and streaming applications. The proposed algorithm first conducts an L1-PCA-informed reliability check to every newly collected point. Then, reliable new points are processed for updating the computed L1-PCs through computationally efficient bit-flipping iterations. Our experimental studies on subspace estimation, image conditioning, and video foreground extraction demonstrated that L1-IPCA attains sturdy outlier resistance, converges fast to the nominal subspace, and has low computational cost.

References

[1] I. T. Jolliffe, *Principal Component Analysis*. New York, NY: Springer, 1986.

[2] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd ed.* New York, NY: John Wiley & Sons, 2012.

[3] C. M. Bishop, *Pattern Recognition and Machine Learning.* New York, NY: Springer-Verlag, 2016.

[4] H. Hotelling, "Analysis of a Complex of Statistical Variables into Principal Components," *J. Edu. Psych.*, vol. 24, pp. 417–441, 1933.

[5] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–218, 1936.

[6] V. Barnett and T. Lewis, *Outliers in statistical data*. New York, NY: Wiley, 1994.

[7] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Optimal algorithms for L1-subspace signal processing," *IEEE Trans. Signal Process.*, vol. 62, pp. 5046–5058, Oct. 2014.

[8] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 30, pp. 1672–1680, Sep. 2008.

[9] M. McCoy and J. A. Tropp, "Two proposals for robust PCA using semidefinite programming," *Electron. J. Statist.*, vol. 5, pp. 1123–1160, Jun. 2011.

[10] F. Nie, H. Huang, C. Ding, D. Luo, and H. Wang, "Robust principal component analysis with non-greedy L1-norm maximization," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI), Barcelona, Spain*, Jul. 2011, pp. 1433–1438.

[11] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. Pados, "Efficient L1-norm principal-component analysis via bit flipping," *IEEE Trans. Signal Process.*, vol. 65, pp. 4252–4264, Aug. 2017.

[12] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, pp. 1–39, May 2011.

[13] P. Narayanamurthy and N. Vaswani, "Provable dynamic robust PCA or robust subspace tracking," *arXiv preprint arXiv:1705.08948*, 2017.

[14] N. Vaswani, T. Bouwmans, S. Javed, and P. Narayanamurthy, "Robust PCA and robust subspace tracking," *arXiv preprint arXiv:1711.09492*, 2017.

[15] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *J. ACM Comput. Surveys (CSUR)*, vol. 35, pp. 399–458, Dec. 2003.

[16] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. IEEE Allerton Conf. Commun. Control Comput., Allerton, IL.* IEEE, Oct. 2010, pp. 704–711.

[17] J. R. Bunch and C. P. Nielsen, "Updating the singular value decomposition," *Numerische Mathematik*, vol. 31, pp. 111–129, 1978.

[18] S. Chandrasekaran, B. Manjunath, Y. F. Wang, J. Winkeler, and H. Zhang, "An eigenspace update algorithm for image analysis," *Graphic. Mod. Imag. Process.*, vol. 59, pp. 321–332, Sep. 1997.

[19] M. Pierantozzi, Y. Liu, D. A. Pados, and S. Colonnese, "Video background tracking and foreground extraction via L1-subspace updates," in *Proc. SPIE CSS*, vol. 9857, Apr. 2016, pp. 985 708–1 – 985 708–16.

[20] Y. Liu and D. A. Pados, "Compressed-sensed-domain L1-PCA video surveillance," *J. IEEE Trans. Mult.*, vol. 18, pp. 351–363, 2016.

[21] D. G. Chachlakis, P. P. Markopoulos, R. J. Muchhala, and A. Savakis, "Visual tracking with L1-Grassmann manifold modeling," in *Proc. SPIE DCS*, vol. 10211-02, april 2017, pp. 1–10.

[22] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Proc. Advan. Neural Inf. Process. Syst. (NIPS).* Vancouver, Canada, 2009, pp. 2080–2088.

[23] G. Mateos and G. B. Giannakis, "Robust PCA as bilinear decomposition with outlier-sparsity regularization," *IEEE Trans. on Signal Process.*, vol. 60, pp. 5176–5190, Oct. 2012.

[24] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the grassmannian for online foreground and background separation in subsampled video," in *Proc. IEEE Conf. Comput. Vis. Patt. Recog. (CVPR).* IEEE, 2012, pp. 1568–1575.

[25] P. P. Markopoulos. L1-PCA code repository. [Online]. Available: https://people.rit.edu/pxmeee/soft.html.

[26] P. P. Markopoulos, S. Kundu, and D. A. Pados, "L1-fusion: Robust linear-time image recovery from few severely corrupted copies," in *Proc. IEEE Int. Conf. Image Process. (ICIP).* IEEE, 2015, pp. 1225–1229.

[27] S. Chamadia and D. A. Pados, "Outlier processing via L1-principal subspaces," *Proc. Florida Artif. Intell. Res. Soc. (FLAIRS)*, pp. 508–513, 2017.

[28] Y. Liu and D. A. Pados, "Conformity evaluation of data samples by l1-norm principal-component analysis," in *Proc. SPIE CSS*, vol. 10658, Apr. 2018, pp. 1 – 9.

[29] Psychological Image Collection at Stirling (PICS). [Online]. Available: http://pics.stir.ac.uk/.

[30] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *ACM Trans. on Graph. (TOG)*, vol. 23. ACM, 2004, pp. 309–314.

[31] B. Lei and L. Q. Xu, "Real-time outdoor video surveillance with robust foreground extraction and object tracking via multi-state transition management," *J. Patt. Recogn. Lett.*, vol. 27, pp. 1816–1825, Nov. 2006.

[32] Context Aware Vision using Image-based Active Recognition (CAVIAR). [Online]. Available: http://homepages.inf.ed.ac.uk/rbf/CAVIAR/.

ISBN 978-90-827970-1-5 © EURASIP 2018