Proceedings of the of the 17th Conference on Computer Science and Intelligence Systems pp. 305–312
DOI: 10.15439/2022F157
ISSN 2300-5963 ACSIS, Vol. 30

Independent Component Analysis Based on Jacobi Iterative Framework and L1-norm Criterion

Adam Borowicz
Faculty of Computer Science, Bialystok University of Technology
Wiejska str. 45A, 15-351 Bialystok, Poland
Email: a.borowicz@pb.edu.pl

***Abstract*—Most recently, a link between principal component analysis (PCA) based on L1-norm and independent component analysis (ICA) has been discovered. It was shown that the ICA can actually be performed by L1-PCA under the whitening assumption, inheriting the improved robustness to outliers. In this paper, a novel ICA algorithm based on Jacobi iterative framework is proposed that utilizes the non-differentiable L1-norm criterion as an objective function. We show that such function can be optimized by sequentially applying Jacobi rotations to the whitened data, wherein optimal rotation angles are found using an exhaustive search method. The experiments show that the proposed method provides a superior convergence as compared to FastICA variants. It also outperforms existing methods in terms of source extraction performance for Laplacian distributed sources. Although the proposed approach exploits the exhaustive search method, it offers a lower computational complexity than that of the optimal L1-PCA algorithm.**

I. Introduction

INDEPENDENT component analysis (ICA) [1] is one of the most widely used techniques in multivariate signal processing. The major goal of the ICA is to transform observed mixtures to components that are as independent from each other as possible. Since the only assumption about the components is that they are mutually independent, the ICA can be viewed as a special case of blind source separation (BBS) problem [2]. Such a problem arises in a wide range of applications, including speech/image source separation, noise reduction, feature extraction, watermark detection.

Most of the ICA algorithms are based on the central limit theorem. Among them, FastICA approach [3], [4], [5] is probably the most well-known example. It attempts to find directions in multidimensional space in which some measure of non-Gaussianity is maximized, thereby enforcing mutual independence between components. The projections of the observed multivariate data onto these directions are viewed as independent components, and often reveal much of the data's structure.

The ICA could also be seen as a generalization of the classical principal component analysis (PCA) [6] by assuming independent and non-Gaussian source distributions. In more detail, the PCA only tries to identify orthogonal directions, along which the data exhibit the greatest variability. Traditionally, this variability is measured using the Frobenius norm (L2-norm on matrices), which allows to decorrelate the components but not to make them independent. Though, the PCA is often used in many ICA algorithms as a pre-processing step for whitening or sphering the data.

In recent years, a growing interest in approaches to the PCA based on the L1-norm can be observed [7], [8], [9]. Unlike conventional PCA, the L1-norm techniques offer an improved robustness to outliers, i.e., data points that differ significantly from the other observations. Most recently, it was shown in [10] that the ICA can actually be performed by L1-norm PCA under the whitening assumption. When the source distribution fulfills certain conditions, it is possible to extract independent components using optimal L1-PCA algorithms with guaranteed global convergence. It was demonstrated that such algorithms may give better accuracy and robustness than those of conventional ICA methods. Unfortunately, optimal L1-PCA algorithms are computationally expensive. In addition, the global convergence is guaranteed only for distributions with negative kurtosis sign. In the work [10], a new variant of the FastICA algorithm with absolute value nonlinearity was considered. Although the accuracy and robustness of this approach were comparable to that of the optimal L1-PCA algorithm, it shows serious convergence difficulties.

In this paper, a novel approach to ICA based on direct optimization of the L1-norm criterion is proposed. The method follows Jacobi iterative framework, whereby the global solution is reached by successively applying Jacobi/Givens rotations to whitened observation vectors [11], [1], [12], [13]. In this way high-dimensional ICA problem is reduced to solving a set of the simpler one-dimensional subproblems. Namely, at every iteration step, we are looking for the angle that maximizes negentropy of the transformed components. Unlike conventional Jacobi methods, the proposed algorithm exploits the contrast function based on the L1-norm, inheriting increased robustness to outliers. Since the local cost functions are of simple form, the optimal rotation angle is found using an exhaustive search method. Therefore, the differentiability of the objective function is no longer required and the method can deal with saddle points and multiple extrema. The simulation results show that the proposed method offers a superior convergence compared to the FastICA method with absolute value function nonlinearity. Furthermore, it outperforms conventional methods in source extraction performance for the mixtures with Laplacian distributions.

This work was supported by Bialystok University of Technology under the grant WZ/WI-IIT/4/2020

The paper is organized as follows. Section II describes the connection between the ICA and L1-norm criterion. It also introduces the mathematical formulations behind the FastICA technique and its recent variant using absolute value function as non-linearity. In Section III, a novel ICA algorithm is proposed based on the Jacobi iterative framework and non-differentiable L1-norm criterion. Section IV investigates the performance of the presented method via numerical simulations. Finally, the conclusions are given in Section V.

II. Link between ICA and L1-PCA

Let us denote by $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^N$ observable, zero-mean N-dimensional random vector and by $\mathbf{y} = [y_1, y_2, \ldots, y_N]^T \in \mathbb{R}^N$ its linear transform, i.e. $\mathbf{y} = \mathbf{B}\mathbf{x}$. Then, the ICA problem consists of finding an unmixing matrix $\mathbf{B} \in \mathbb{R}^{N \times N}$ such that the components of $\mathbf{y}$ are as independent as possible. Since statistical independence implies uncorrelatedness, many ICA algorithms assume explicitly that $E\{\mathbf{y}\mathbf{y}^T\} = \mathbf{I}$, where $E\{.\}$ stands for expectation operator. This is usually enforced by the following factorization of the unmixing matrix:

$$\mathbf{B} = \mathbf{W}\mathbf{C}_{\mathbf{xx}}^{-1/2}, \tag{1}$$

where $\mathbf{C}_{\mathbf{xx}}^{-1/2}$ denotes the whitening matrix which is computed by inverting a square root of the observation signal covariance matrix. It can be easily verified that the constraint $E\{\mathbf{y}\mathbf{y}^T\} = \mathbf{I}$ holds for any orthogonal matrix $\mathbf{W}$. Since the whitening transformation is always possible, the ICA problem can be viewed as finding an orthogonal transformation of the whitened data vector $\mathbf{z} = \mathbf{C}_{\mathbf{xx}}^{-1/2}\mathbf{x}$, i.e. $\mathbf{y} = \mathbf{W}\mathbf{z}$, such that the components of $\mathbf{y}$ are as independent as possible. This can in particular be achieved by maximizing the negentropy of the random vector $\mathbf{y}$ defined as follows:

$$J(\mathbf{y}) = h(\mathbf{v}) - h(\mathbf{y}), \tag{2}$$

where $h(.)$ is the differential entropy [14] and $\mathbf{v}$ is the Gaussian random variable of the same covariance matrix as $\mathbf{y}$. Unfortunately, computation of (2) is not an easy task and in practice some approximations of negentropy [3], [5] have to be used. Let $y = \mathbf{w}^T\mathbf{z}$ denote a random variable being a linear projection of the whitened data vector onto some direction $\mathbf{w} \in \mathbb{R}^N$. Then, the negentropy of this projection can be approximated as follows:

$$J_g(y) \approx c[E\{g(y)\} - E\{g(v)\}]^2, \tag{3}$$

where c is irrelevant constant and g is any non-quadratic, sufficiently smooth even function. The variables v, y are assumed to be of zero mean and unit variance, with v being a Gaussian-distributed variable. The approximation (3) is interpreted as a measure of non-Gaussianity as it is always non-negative, and it equals to zero if and only if y is Gaussian.

In the case of the deflationary FastICA algorithm [4], the independent components are found sequentially, one after another. For each source, the criterion (3) is optimized iteratively, using an approximate Newton technique. Namely, the following fixed-point iteration is used:

$$\hat{\mathbf{w}} = E\{\mathbf{z}g'(\mathbf{w}^T\mathbf{z})\} + E\{g''(\mathbf{w}^T\mathbf{z})\}\mathbf{w}, \tag{4}$$

$$\mathbf{w}^+ = \hat{\mathbf{w}}/\|\hat{\mathbf{w}}\|_2, \tag{5}$$

where $\mathbf{w}^+$ stands for the direction vector of the estimated independent component after the current iteration. These vectors are projected onto the space orthogonal to the space spanned by the earlier found vectors, so that at the end, we obtain the set $\{\mathbf{w}_i^T\}_{i=1}^N$ of orthogonal projectors that are stored in the rows of the matrix $\mathbf{W}$.

A. FastICA based on absolute value function

A crucial step in optimizing the FastICA algorithm is to choose the best non-linearity $g(.)$ [15]. Many ICA algorithms [11], [1], [12], [16] use kurtosis-based contrast functions, which correspond to the fourth-power non-linearity $g(y) = 1/4y^4$. Such a choice can be justified on statistical grounds only for estimating sub-Gaussian sources (i.e. those with negative kurtosis) when there are no outliers. However, in practice we mostly deal with super-Gaussian variables [17] that have positive kurtosis. It was suggested in [3], [4] that for super-Gaussian densities, the optimal contrast function is a function that grows slower than quadratically. In particular, as a general-purpose contrast function, one should choose,

$$g(y) = |y|^\alpha, \quad \alpha < 2. \tag{6}$$

Nevertheless, no attempt has been made to implement this idea in practice. The reason is that the FastICA algorithm assumes the differentiability of $g(y)$, whereas for the absolute value function, this property fails at origin. Therefore, the following differentiable approximation of the absolute value function has been proposed:

$$g(y) = \frac{1}{a}\log\cosh(ay), \tag{7}$$

with $1 \leq a \leq 2$. However, this approximation may not provide the same independent source extraction performance as the absolute value function.

In the recent work [10], the authors admitted differentiability of $g(y) = |y|$ by assuming that $g'(y) = \text{sign}(y)$ and $g''(y) = 2\delta(y)$, where $\delta(y)$ denotes Dirac's delta function. It resulted in the modified FastICA method with the following iteration step in place of (4):

$$\hat{\mathbf{w}} = E\{\mathbf{z}\,\text{sign}(\mathbf{w}^T\mathbf{z})\} - 2f_y(0)\mathbf{w}, \tag{8}$$

where $f_y(0)$ stands for the probability density function (PDF) of y evaluated at the origin. As proposed in [10], it can be computed through the kernel density estimate with Gaussian kernel. It was also demonstrated that, for small data sizes (e.g. $N = 2, 3$), this algorithm can provide some improvements in source extraction performance when dealing with outliers. However, the iteration (8) may present difficulties converging to the right solution. Please note that, at each iteration, the PDF of the extractor output must be estimated, which may be impractical.

B. ICA via L1-PCA

Let $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_M] \in \mathbb{R}^{N \times M}$ denotes data matrix, where $\{\mathbf{z}_m\}_{m=1}^{M}$ are realizations of a zero-mean random vector $\mathbf{z}$. Assuming ergodicity conditions and that $g(y) = |y|$, it can be shown that for large enough sample size the L1-norm of the projection $\mathbf{w}^T\mathbf{Z}$ becomes proportional to $E\{g(y)\}$,

$$\|\mathbf{w}^T\mathbf{Z}\|_1 = \sum_{m=1}^{M} |\mathbf{w}^T\mathbf{z}_m| \to ME\{g(y)\}. \tag{9}$$

Please note that the second term in (3) is always constant. Thus, the solution is reached at a certain optimum (i.e. maximum or minimum) of $E\{g(y)\}$ under the constraint $\|\mathbf{w}\|_2 = 1$. For this reason, the ICA can also be accomplished by whitening, followed by the minimization or maximization of the L1-norm. It was shown in [10] that symmetric sources with negative (respectively, positive) kurtosis are maximizers (respectively, minimizers) of $E\{|y|\}$. Whereas, the optimization problem of finding L1 principal component can be formulated as follows [8], [9]:

$$\mathbf{w}_{L1} = \underset{\mathbf{w}\in R^N, \|\mathbf{w}\|_2=1}{\operatorname{argmax}} \|\mathbf{w}^T\mathbf{Z}\|_1. \tag{10}$$

Since the objective function is non-differentiable, the problem is difficult to solve by means of conventional optimization techniques such as gradient-based methods. However, it was shown in [8] that $\mathbf{w}_{L1} = \mathbf{Z}\mathbf{c}_{\text{opt}}/\|\mathbf{Z}\mathbf{c}_{\text{opt}}\|_2$, where

$$\mathbf{c}_{\text{opt}} = \underset{\mathbf{c}\in\{\pm 1\}^M}{\operatorname{argmax}} \|\mathbf{Z}\mathbf{c}\|_2. \tag{11}$$

Hence, the L1-norm maximization can be viewed as a combinatorial problem over the binary field. A globally convergent L1-PCA algorithm with complexity $\mathcal{O}(M^{\text{rank}(\mathbf{Z})})$ was proposed in [8]. A faster, yet suboptimal version of this approach [9] is based on consecutive bit-flipping operations. Though, its time complexity can still be prohibitive for large data sizes. The most computationally efficient L1-PCA method was proposed earlier in [7]. It is based on the fixed-point iterative scheme similar to that used in FastICA algorithm. Unfortunately, the method often gets trapped in local extrema. Despite these shortcomings, the L1-PCA algorithms can be used directly to extract independent sources with negative kurtosis sign (i.e. sub-Gaussians) under whitening assumption. It was shown in [10] that globally convergent L1-PCA algorithm may give better accuracy and robustness than those of the conventional ICA methods, especially when dealing with outliers. The L1-PCA algorithm can also be modified to perform L1-norm minimization. However, in such case the global convergence property is lost because the L1-norm and the L2-norm minimization problems are not related as in (10)-(11). In addition, computational complexity of this algorithm can become prohibitive for large sample size and/or observation dimensions. In most applications, only suboptimal L1-PCA algorithms [9], [7] can be considered.

III. Proposed method

The proposed method is based on the Jacobi iterative framework [12]. Namely, an objective function is optimized by applying successively orthogonal transformations to the whitened observation data vectors:

$$\mathbf{y}^{(k+1)} = \mathbf{G}(p_k, q_k, \theta_k)\mathbf{y}^{(k)}, \quad k = 1, 2, ..., \tag{12}$$

where $\mathbf{y}^{(1)} = \mathbf{z} = \mathbf{C}_{\mathbf{xx}}^{-1/2}\mathbf{x}$. The matrix $\mathbf{G}(p, q, \theta)$ represents Jacobi rotation [18] by the angle θ in the plane determined by the p and q coordinates, i.e.:

$$\mathbf{G}(p,q,\theta) = \begin{bmatrix} \mathbf{I}_{p-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \cos\theta & \mathbf{0} & \sin\theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{q-p-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\sin\theta & \mathbf{0} & \cos\theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{N-q-1} \end{bmatrix}, \tag{13}$$

with $1 \le p < q \le N$, Thus, the unmixing matrix after the kth iteration can be expressed as follows:

$$\hat{\mathbf{B}}^{(k)} = \left(\prod_{i=1}^{\overset{\curvearrowleft}{k}} \mathbf{G}(p_i, q_i, \theta_i) \right) \mathbf{C}_{\mathbf{xx}}^{-1/2}. \tag{14}$$

Please note that for N-dimensional space we have $N(N-1)/2$ possible rotation planes, each uniquely represented by pair (p, q). A sequence of rotations represented by these pairs is arranged in a so-called sweep. In fact, any rotation order is allowed, but some may work better than others [19], [20]. In this work, a typical row-cycling ordering is used as described in Tab. I. Usually, it is necessary to go through several sweeps before convergence is achieved. The algorithm is terminated when, for all rotations in the current sweep, we have $|\theta_k| < \theta_{\min}$, or when the maximum number of sweeps is reached. The parameter $\theta_{\min}$ is an empirically chosen small angle [12], which controls the accuracy of the optimization.

It is crucial for this estimation framework to compute the rotation angles θ_k so that a given objective function is gradually optimized. Motivated by the ideas presented in the previous section, we propose to maximize negentropy approximation (3) with $g(y) = |y|$ directly. Since we deal with a sequence of two-dimensional ICA problems, the objective function for two units must be considered. As suggested in [4], such a function can be defined as the sum of the one-unit functions:

$$J^{(k)}(\theta) = \sum_{i\in\{p_k, q_k\}} |E\{|\hat{y}_i^{(k)}(\theta)|\} - E\{|v|\}|, \tag{15}$$

where

$$\hat{y}_{p_k}^{(k)}(\theta) = y_{p_k}^{(k)} \cos\theta + y_{q_k}^{(k)} \sin\theta, \tag{16}$$

TABLE I: Arrangement of rotation planes using a row-cycling ordering for $N = 3$.

sweep no.	1			2			...
k	1	2	3	4	5	6	...
(p_k, q_k)	(1,2)	(1,3)	(2,3)	(1,2)	(1,3)	(2,3)	...

$$\hat{y}_{q_k}^{(k)}(\theta) = y_{q_k}^{(k)} \cos\theta - y_{p_k}^{(k)} \sin\theta, \tag{17}$$

are respectively the p_kth and q_kth coefficient of the currently transformed data vector $\mathbf{y}^{(k)}$. Please note that for a normally distributed random variable v with mean μ and variance σ^2, the random variable $u = |v|$ has a folded normal distribution. The mean of the folded distribution is given by [21]:

$$\mu_u = \sigma\sqrt{2/\pi}\exp\left(-\frac{\mu^2}{2\sigma^2}\right) + \mu\,\mathrm{erf}\left(\frac{\mu}{\sqrt{2\sigma^2}}\right) \tag{18}$$

For $\mu = 0$ and $\sigma^2 = 1$, the expectation $E\{|v|\}$ in (15) reduces to the constant factor $\sqrt{2/\pi}$. Examples of the objective function (15) evaluated at 12 consecutive data rotations are presented in Fig. 1. As we see, these functions are always periodic with period $\pi/2$. Therefore, the search for the optimal angle can be restricted to the interval $[-\pi/4; \pi/4)$, i.e.:

$$\hat{\theta}_k = \underset{-\pi/4 \le \theta < \pi/4}{\mathrm{argmax}} \; J^{(k)}(\theta). \tag{19}$$

In order to construct Newton-type iteration scheme, one can admit differentiability of $g(y) = |y|$ in a similar way as for the FastICA approach. In Fig. 1 we see that the objective function contains multiple local maxima and saddle points, thus even if we use a differentiable approximation for the absolute value function, it would be difficult to reach the global maximum of (15). However, in this case, each plane rotation depends on a single parameter θ_k, reducing the N-dimensional optimization problem to the sequence of the $N(N-1)/2$ one-dimensional search subproblems per sweep. Therefore, as opposed to the FastICA approach, the solution can be found using an exhaustive method in a reasonable execution time. In particular, for each data rotation, the function (15) can be evaluated at the set of D equidistant points:

$$\theta \in \{-\pi/4 + i\pi/(2D) : i = 0, 1, ..., D-1\}. \tag{20}$$

The greater the value of D, the better the accuracy of the optimization. For even D, the set (20) always contains a zero value. Hence, in order to ensure local convergence, the parameter $\theta_{\min}$ for stop condition should be set to any value in the interval $(0; \pi/(2D))$. We have found empirically that the rotation angles tend to decrease in subsequent sweeps, and some optimizations are possible. For example, it may not be necessary to search for optimal angle over entire interval at the later sweeps. The exhaustive search algorithm can also be replaced by more sophisticated non-gradient techniques such as simplex bisection method [22], particle swarm optimization [23], genetic algorithms [24], simulated annealing [25].

The Matlab implementation of the proposed approach is given in Alg. 1. The expectations in (15), are replaced by sums with observables in place of random variables. Please note that since the matrix $\mathbf{G}(p, q, \theta)$ modifies only (p, q) rows, it is not necessary to compute it explicitly. It is easy to see that in each sweep, we must perform $N(N-1)/2$ data rotations. Each rotation costs $4M$ multiplications, but this operation must be repeated D times as the objective function is evaluated at D points. Thus, the time complexity of the single sweep can be roughly estimated as of order $\mathcal{O}(N^2MD)$.

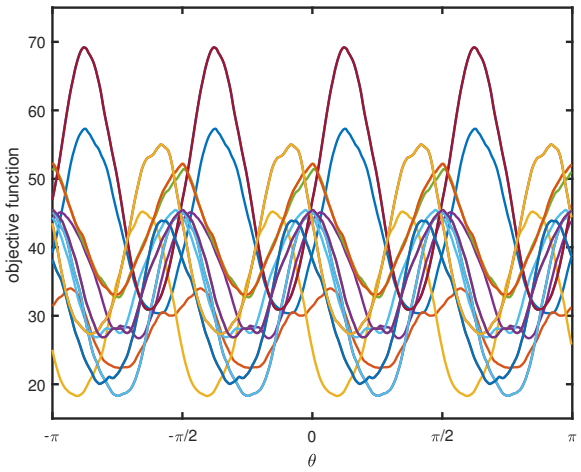


Fig. 1: Examples of the function (15) evaluated at 12 consecutive data rotations for a randomly generated mixture of $N = 4$ Laplacian distributed sources with sample size $M = 400$.

Algorithm 1 Matlab implementation of the proposed method

```
function [Y,B,k] = JICA_abs(X, k_max, D)
[N, M] = size(X);
X = X - mean(X, 2);
B = inv(sqrtm((X*X')/M));
Y = B*X;
D = D + mod(D, 2);
theta = linspace(-pi/4, pi/4-pi/2/D, D);
c = cos(theta)';
s = sin(theta)';
Gp = [  c s ];
Gq = [ -s c ];
mu = M * sqrt(2/pi);
k = 1; encore = 1;
while k <= k_max && encore
    encore = 0;
    for p = 1:N-1
      for q = p+1:N
        r = [p q];
        Yp = Gp*Y(r,:);
        Yq = Gq*Y(r,:);
        J = abs(sum(abs(Yp), 2)-mu) +...
            abs(sum(abs(Yq), 2)-mu);
        [~, I] = max(J);
        if abs(theta(I)) > pi/4/D
          encore = 1;
          Y(r,:)=[Yp(I,:);Yq(I,:)];
          B(r,:)=[Gp(I,:);Gq(I,:)]*B(r,:);
        end
      end
    end
    k = k + 1;
end
```

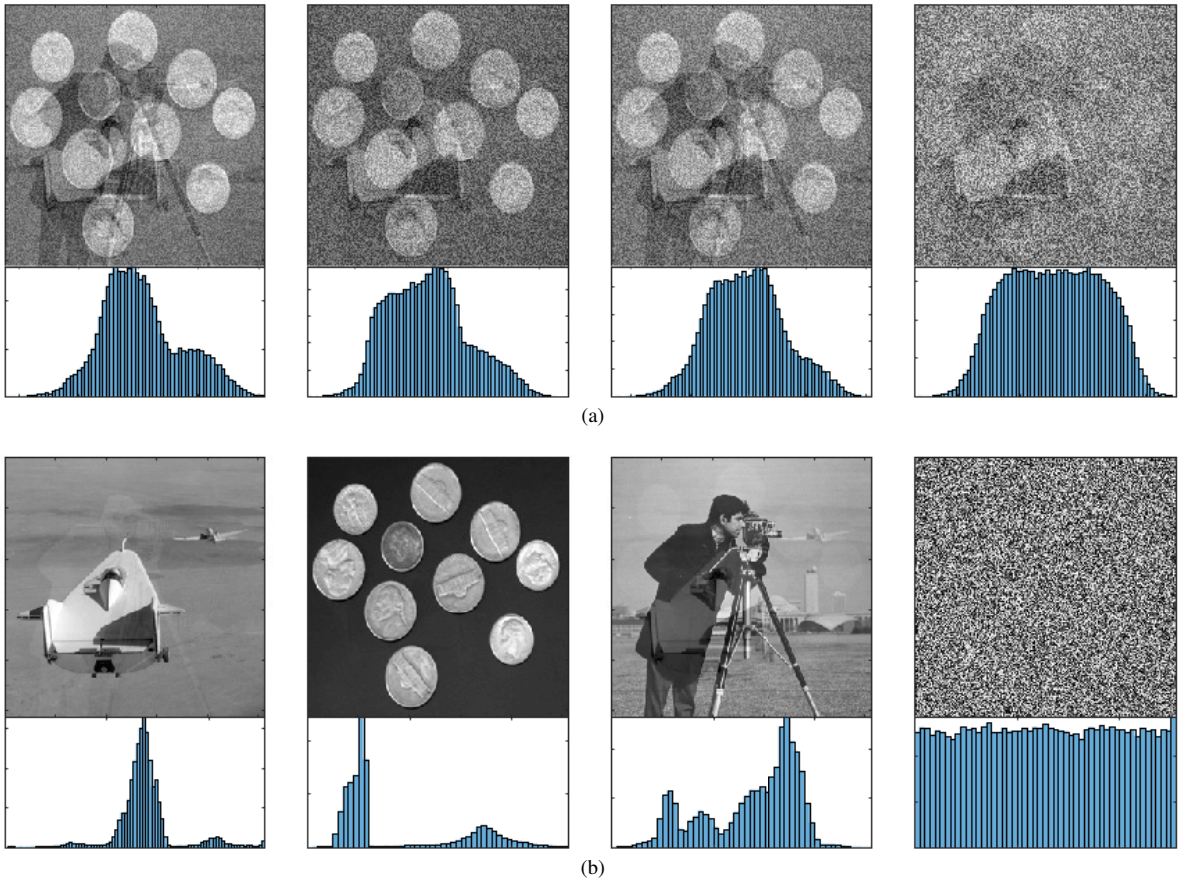


Fig. 2: Separation of the images with various distributions of pixel gray levels. (a) Randomly mixed images and histograms. (b) Recovered images and histograms.

IV. Experiments

A. Illustrative examples

As a toy example, we considered a mixture of three Matlab built-in images and uniform noise. The images were resized to the same size 256×256 pixels with an 8-bit grayscale. The mixtures and their histograms are depicted in Fig. 2a. As can be seen, the source images have been significantly degraded, but also the form of the mixed data histograms is more like the Gaussian function. Fig. 2b shows the images recovered using the proposed method and the corresponding histograms. This example clearly shows that the algorithm is capable of transforming data from normality to independent marginal distributions.

B. Independent source extraction performance

In this experiment, the source extraction performance of the proposed algorithm is evaluated. In order to distinguish the algorithm from the existing techniques, it was denoted by the acronym JICA-abs, which stands for "Jacobi-type ICA based on absolute value function." Also, several existing techniques including state-of-art methods were chosen for comparison, namely: joint approximate diagonalization of eigenmatrices (JADE) [11], the conventional FastICA algorithms with the fourth-power non-linearity (FastICA-4power) and the differentiable approximation of the absolute value (FastICA-logcosh) [4], the modified FastICA algorithm based on direct use of the absolute value criterion (FastICA-abs) [10], the iterative L1-PCA [7] and more accurate bit-flipping L1-PCA method (L1-BF) [9]. In this comparison, we do not consider the optimal L1-PCA algorithm [8] due to high computational demands. On the other hand, it was shown in [10] that for the sources with uniform densities, the source extraction performance of the iterative L1-PCA method is similar to that of the optimal algorithm.

It is rather common that the performance of an iterative algorithm may vary depending on the stop conditions and initialization. Therefore, in all methods, the matrix $\mathbf{W}$ was initialized to the identity matrix. The FastICA and iterative L1-PCA algorithms were stopped when for all sources the following condition was met: $1 - |\mathbf{w}^T\mathbf{w}^+| < \epsilon$, or when a maximum number of 1000 iterations was reached. For all these methods, except FastICA-abs, the ϵ parameter was set to 10^{-4}. In the case of the FastICA-abs, it was necessary to increase its value to 10^{-3} due to convergence difficulties. For the JICA-abs and JADE methods, the reliable and stable results were obtained when the maximum number of sweeps was set to 20

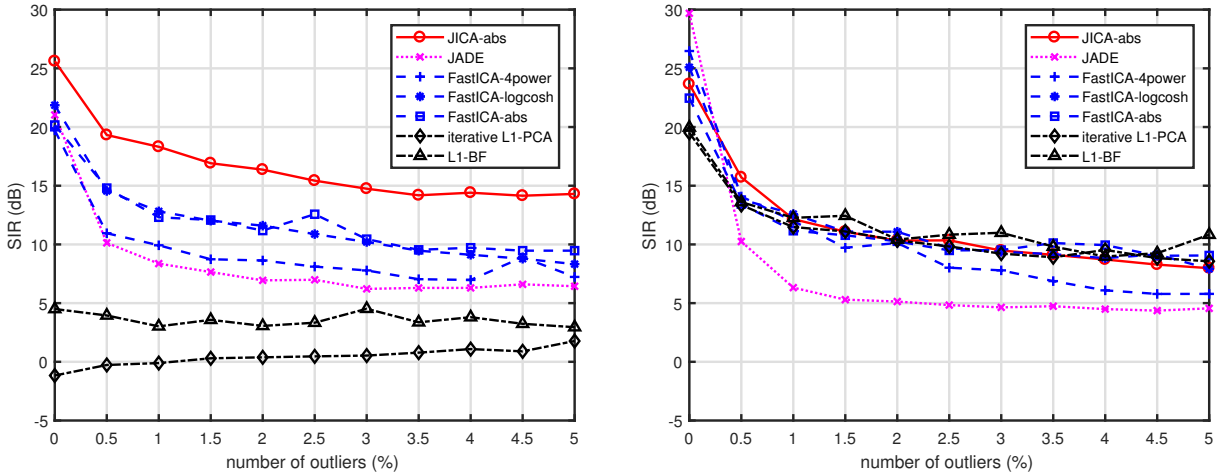


Fig. 3: Independent source extraction performance as a function of the outlier contamination rate, for $N = 4$ sources with Laplacian distribution (left), and uniform distribution (right). The length of source signals is fixed at $M = 400$ samples.

and the angle $\theta_{\min} = \pi/(4D)$ with $D = 128$. We verified empirically that rotations at a smaller angle than $\pi/512$ are not statistically significant.

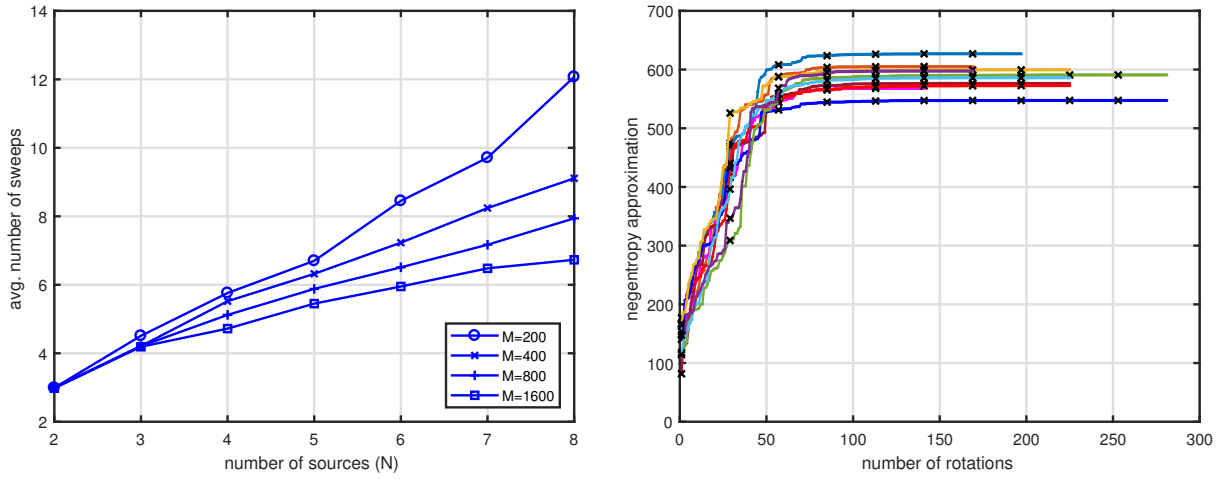
For comparative purposes, we considered the linear mixtures of $N = 4$ synthetic sources, all with negative or positive kurtosis sign generated from uniform and Laplacian distribution, respectively. The length of each source signal was set to $M = 400$ samples. The coefficients of the mixing matrix were generated from the uniform distribution. Ill-conditioned matrices (with a condition number greater than 100) were excluded from the evaluation. In order to evaluate the robustness of the algorithms against outliers, randomly chosen observations were replaced with noise spikes drawn from a Gaussian distribution $\mathcal{N}(10, 1)$ at varying contamination rates.

The independent source extraction performance was estimated using the average signal-to-interference ratio (SIR) [26], [27], [28]. Please note that the higher the value of the SIR is, the better performance we get. The performance indexes were averaged over 1000 random realizations of the sources and the mixing matrices, but at each Monte Carlo run, all methods were operating on the same data. Fig. 3 presents the source extraction performance of different algorithms for various percentages of outliers. As can be seen, the proposed method clearly outperforms existing algorithms on average by 5dB for Laplacian distributed sources. In this case, the iterative L1-PCA algorithm provides the worst performance, as it is designed to only maximize L1-norm, whereas for distributions with positive kurtosis sign the L1-norm should be minimized. For Laplacian distributed sources, the L1-BF algorithm was modified to minimize the L1-norm according to the suggestion in [10]. In result, the source extraction performance of this method is slightly better than that of the iterative L1-PCA algorithm. Though, it is still poor compared to the ICA approaches. This confirms our earlier remark that the L1-norm and L2-norm minimization problems are not related in the same way as the corresponding maximization problems (10)-(11). Although there is no clear winner for uniformly distributed sources, the JADE and FastICA-4power methods provide the best performance in the absence of outliers. Clearly, the absolute value criterion may not be the best choice for sub-Gaussian distributed sources. On the other hand, the approaches, whether based on absolute value criterion or on differentiable approximations thereof, show increased robustness to outliers as compared to the kurtosis-based methods.

C. Convergence and execution time

The total execution time of an iterative algorithm depends on the convergence rate. Unfortunately, rigorous convergence analysis of the proposed approach is not an easy task and is out of the scope of this paper. Though, we measured the average number of iterations (sweeps) taken by the presented algorithm until convergence was reached for various data sizes. The results averaged over 1000 independent runs are depicted in Fig. 4a. As can be seen that the number of iterations increases with the number of sources, but this dependency is weaker than linear, for sufficiently large M. It is rather not surprising, because as the sample size increases, an objective function usually becomes smoother and thus a faster convergence can be achieved. In this case, the algorithm converges to the stationary solution in a relatively small number of sweeps. However, please note that each sweep consists of $N(N-1)/2$ data rotations. In order to better illustrate the convergence properties of the algorithm, in Fig. 4b, we also show the global cost function measured after each data rotation for 10 independent Monte Carlo runs. It is rather clear that the algorithm converged quickly in all cases.

In order to compare the computational complexity of the proposed algorithm with the existing methods, we measured their execution times. The experiments were carried out in the Matlab environment, running on AMD Ryzen 5 3550H processor. Tab. II presents the minimum, maximum, and



(a) (b)

Fig. 4: Evaluation of the convergence properties of the proposed method. (a) Average number of sweeps. (b) Global contrast functions measured after each data rotation in 10 independent Monte Carlo runs. It was computed as a sum of the approximations (3) with $g(y) = |y|$ for all rows of the data matrix. The first data rotation in each sweep is denoted by cross mark.

TABLE II: Execution times (in milliseconds) and percentage of runs where the maximum number of iterations is reached in the experiments of Fig. 3 without outliers.

	Laplacian				uniform			
algorithm	$t_{\min}$	$t_{\max}$	t_{avg}	failures (%)	$t_{\min}$	$t_{\max}$	t_{avg}	failures (%)
JICA-abs	3.99	17.04	8.47	0	5.43	26.15	9.02	0
JADE	0.69	9.94	1.09	0	0.78	10.73	1.20	0
FastICA-4power	0.30	18.41	0.53	0.4	0.30	3.52	0.44	0
FastICA-logcosh	0.46	34.00	0.84	0.3	0.45	1.82	0.69	0
FastICA-abs	0.60	120.26	38.98	62.3	0.53	86.90	6.66	9.5
iter. L1-PCA	0.21	2.94	0.42	0	0.19	2.84	0.37	0
L1-BF	23.36	37.23	28.62	0	27.82	72.35	42.94	0

TABLE III: Execution times (in milliseconds) and percentage of runs where the maximum number of iterations is reached in the experiments of Fig. 3 with 5 percent of outliers.

	Laplacian				uniform			
algorithm	$t_{\min}$	$t_{\max}$	t_{avg}	failures (%)	$t_{\min}$	$t_{\max}$	t_{avg}	failures (%)
JICA-abs	3.72	18.44	8.88	0	5.37	25.12	9.07	0
JADE	0.70	10.88	1.24	0	0.68	10.54	0.99	0
FastICA-4power	0.30	16.22	0.55	0.1	0.30	18.32	1.52	5.7
FastICA-logcosh	0.47	33.86	1.16	0.3	0.51	63.12	4.51	9.4
FastICA-abs	0.52	111.34	26.01	59.8	0.62	109.39	39.93	82.1
iter. L1-PCA	0.23	3.38	0.36	0	0.24	3.59	0.36	0
L1-BF	24.80	39.64	30.34	0	26.64	81.41	46.48	0

average execution times collected in the experiment of Fig. 3 for data without outliers. The columns denoted as "failures" show the percentage of the Monte Carlo runs where the maximum iteration number was reached. The same statistics are presented in Tab. III, but for data with 5 percent of outliers. Although the JICA-abs method has relatively long average execution time, it is much faster than L1-BF algorithm. The proposed method also provides better convergence properties than those of the FastICA-based algorithms. Similarly to the JADE method and approximate L1-PCA algorithms, the JICA-abs approach always converged to a stationary solution within the iteration limit. The FastICA-4power and FastICA-logcosh methods sometimes reach the iteration limit, which results in the increased maximum execution time. It is especially evident for uniformly distributed sources with outliers, where these methods reach the iteration limit in around 6-9 percent of runs. Unfortunately, the FastICA-abs method present even more serious convergence difficulties when dealing with outliers. In this case, the iteration limit is reached in around 82 and 60 percent of runs, for uniform and Laplacian distributions,

respectively. Obviously, the upper bound of the execution time can be reduced by decreasing the iteration limit, but it can also deteriorate the accuracy of the optimization. Therefore, a method with a smaller upper bound of the execution time may be a better choice when the timeliness of the system becomes a prominent problem.

V. Conclusion

A novel ICA algorithm has been proposed that directly utilizes non-differentiable absolute value criterion as a contrast function for the ICA problem. The algorithm is based on Jacobi iterative framework and exhaustive search method. Experimental studies show that the proposed approach provides better accuracy and robustness to outliers than existing methods for Laplacian distributed sources. Unlike the FastICA approaches, it does not show any convergence issues. Though, it has on average relatively high execution time as compared to the state-of-art ICA methods. On the other hand, it is faster than currently most accurate suboptimal L1-PCA algorithm that also works in an exhaustive manner.

A rigorous convergence analysis of the proposed method is of great theoretical importance, thus it should be the subject of further research. We also believe that the computational complexity can potentially be reduced. In addition, future works may include developing practical applications in speech, audio and image denoising.

References

[1] P. Comon, "Independent component analysis, a new concept?" *Signal Process.*, vol. 36, no. 3, pp. 287–314, 1994. doi: 10.1016/0165-1684(94)90029-9

[2] P. Comon and C. Jutten, Eds., *Handbook of Blind Source Separation. Independent Component Analysis and Applications*, 1st ed. Oxford, USA: Academic Press, inc., 2010.

[3] A. Hyvärinen, "New approximations of differential entropy for independent component analysis and projection pursuit," in *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems 10*, 1997, pp. 273–279.

[4] ——, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. Neural Netw.*, vol. 10, no. 3, pp. 626–634, 1999. doi: 10.1109/72.761722

[5] ——, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, 07 1999.

[6] I. T. Jolliffe, *Principal Component Analysis*. New York: Springer Verlag, 2002.

[7] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 9, pp. 1672–1680, 2008. doi: 10.1109/TPAMI.2008.114

[8] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Optimal algorithms for L_1-subspace signal processing," *IEEE Transactions on Signal Processing*, vol. 62, no. 19, pp. 5046–5058, 2014. doi: 10.1109/TSP.2014.2338077

[9] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. A. Pados, "Efficient L1-norm principal-component analysis via bit flipping," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4252–4264, 2017. doi: 10.1109/TSP.2017.2708023

[10] R. Martın-Clemente and V. Zarzoso, "On the link between L1-PCA and ICA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 515–528, 2017. doi: 10.1109/TPAMI.2016.2557797

[11] J. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *IEE Proceedings F - Radar and Signal Processing*, vol. 140, no. 6, pp. 362–370, 1993. doi: 10.1049/ip-f-2.1993.0054

[12] J. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation*, vol. 11, no. 1, pp. 157–192, 1999. doi: 10.1162/089976699300016863

[13] E. Learned-Miller and J. Fisher, "ICA using spacings estimates of entropy," *Journal of Machine Learning Research*, vol. 4, pp. 1271–1295, Dec. 2003.

[14] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006.

[15] A. Dermoune and T. Wei, "FastICA algorithm: Five criteria for the optimal choice of the nonlinearity function," *IEEE Transactions on Signal Processing*, vol. 61, pp. 2078–2087, 04 2013. doi: 10.1109/TSP.2013.2243440

[16] V. Zarzoso and P. Comon, "Robust independent component analysis by iterative maximization of the kurtosis contrast with algebraic optimal step size," *IEEE Transactions on Neural Networks*, vol. 21, no. 2, pp. 248–261, 2010. doi: 10.1109/TNN.2009.2035920

[17] A. Bell and T. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995. doi: 10.1162/neco.1995.7.6.1129

[18] G. Golub and C. Van Loan, *Matrix Computations*. USA: Johns Hopkins University Press, 2013.

[19] W. Ouedraogo, A. Souloumiac, and C. Jutten, "Non-negative independent component analysis algorithm based on 2D Givens rotations and a Newton optimization," in *Latent Variable Analysis and Signal Separation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. doi: 10.1007/978-3-642-15995-4 pp. 522–529.

[20] M. Parfieniuk, "A parallel factorization for generating orthogonal matrices," in *International Conference on Parallel Processing and Applied Mathematics (PPAM) 2019*. Bialystok, Poland: Springer, 2019. doi: 10.1007/978-3-030-43229 pp. 567–578.

[21] M. Tsagris, C. Beneki, and H. Hassani, "On the folded normal distribution," *Mathematics*, vol. 2, no. 1, pp. 12–28, feb 2014. doi: 10.3390/math2010012

[22] C. Samuelsson, "Comparative evaluation of the stochastic simplex bisection algorithm and the scipy.optimize module," in *Proceedings of the 2015 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 5, 2015. doi: 10.15439/2015F47 pp. 573–578.

[23] T. Krzeszowski and K. Wiktorowicz, "Evaluation of selected fuzzy particle swarm optimization algorithms," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 8, 2016. doi: 10.15439/2016F206 pp. 571–575.

[24] K. Pytel, "Hybrid multievolutionary system to solve function optimization problems," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 11, 2017. doi: 10.15439/2017F85 pp. 87–90.

[25] A. Alihodzic, S. Delalić, and D. Gusic, "An effective integrated metaheuristic algorithm for solving engineering problems," in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 21, 2020. doi: 10.15439/2020KM81 pp. 207–214.

[26] P. Tichavsky, Z. Koldovsky, and E. Oja, "Performance analysis of the FastICA algorithm and Cramér-Rao bounds for linear independent component analysis," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1189–1203, 2006. doi: 10.1109/TSP.2006.870561

[27] Z. Koldovský, P. Tichavsky, and E. Oja, "Efficient variant of algorithm FastICA for independent component analysis attaining the Cramér-Rao lower bound," *IEEE Transactions on Neural Networks*, vol. 17, pp. 1265–77, 10 2006. doi: 10.1109/TNN.2006.875991

[28] A. Borowicz, "Orthogonal approach to independent component analysis using quaternionic factorization," *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 39, p. 23, September 2020. doi: 10.1186/s13634-020-00697-0