ICA with Reference

Wei Lu [a] and **Jagath C. Rajapakse** [b]

[a] Sony Singapore Research Laboratory, Singapore

[b] School of Computer Engineering, Nanyang Technological University, Singapore

15th June 2005

Corresponding Author :

Jagath C. Rajapakse, Ph.D.

School of Computer Engineering

Nanyang Technological University

Block N4, Nanyang Avenue

Singapore, 639798

phone: +65 6790 5802

fax: +65 6792 6559

email: asjagath@ntu.edu.sg

Abstract

We present the technique of the ICA with Reference (ICA-R) to extract an interesting subset of independent sources from their linear mixtures when some a priori information of the sources are available in the form of rough templates (references). The constrained Independent Component Analysis (cICA) is extended to incorporate the reference signals that carry some information of the sources as additional constraints into the ICA contrast function. A neural algorithm is then proposed using a Newton-like approach to obtain an optimal solution to the constrained optimization problem. Stability of the convergence and selection of parameters in the learning algorithm are analyzed. Experiments with synthetic signals and real fMRI data demonstrate the efficacy and accuracy of the proposed algorithm.

Keywords: *constrained ICA (cICA), constrained optimization, functional MRI, Independent Component Analysis (ICA), ICA with Reference (ICA-R), non-Gaussianity.*

1 Introduction

Some applications of Blind Source Separation (BSS) often wish to extract a desired subset of sources and automatically discard uninteresting sources, such as artifacts and noises, from the observed mixtures: for example, the extraction of the curves showing intrinsic market variations in financial analysis, or the separation of speakers' voices from the environmental noises in audio applications [1, 2]. Conventionally, the second-order methods, like the Minimum Mean Square Error (MMSE) technique, were often used to detect, extract, and recognize the desired signals [3], but their applications were limited because of the usage of only second-order statistics. In the last decade, Independent Component Analysis (ICA) has become a popular method, as a higher-order statistical technique, to provide solutions to the BSS problems [4, 5, 6]. As the classical ICA algorithms always extract independent components (ICs) which number is same as the number of the observed mixtures, additional post-processing is required to select the desired sources from the set of recovered sources. Such a two-stage method involves redundant computation, requires large memory for estimating unnecessary signals, and degrades the quality of the signals recovered, especially in high-dimensional applications.

Recently, some ICA algorithms, like nonlinear PCA [7] and fastICA [8], have been proposed to separate a single or a subset of independent sources from their mixtures. However, the extraction of signals at the global optimum by using these algorithms is always determined by the contrast function adopted. Furthermore, the recovered subset of sources is sometimes arbitrary due to the existence of local minima, theoretically, caused by the energy minimizing techniques used. Some researchers have incorporated additional information by using sparse decomposition of signals [9], fourth-order cumulants [10] or filter banks [11] into the ICA contrast function to find the global optimum. However, such algorithms still suffer from the limitation that a particular statistic should be considered for the extraction, which determines the sources extracted.

On the other hand, additional knowledge about the sources or the mixing channels are available in some BSS applications, for example, statistical properties of audio signals or physical distances between the locations of the microphones in the cocktail-party problem. These can be treated as *a priori* information to the ICA contrast functions to facilitate more applications and improve the accuracy of the ICA. Here, we consider a particular

instance where a priori information is available as traces of the interesting sources. A good example is the ON-OFF input stimuli in functional Magnetic Resonance Imaging (fMRI) experiments [12, 13]. The rough templates of the desired sources, available from a priori information, are referred to as the *reference* signals. These reference signals carry adequate information to distinguish the desired sources from artifacts and noises, when incorporated into the ICA algorithm, but are not identical to the corresponding original sources.

This manuscript proposes a variation to the classical ICA paradigm to extract one or several desired independent sources from a set of observations that are linear mixtures of the sources by incorporating the reference signals into the contrast function. The constrained Independent Component Analysis (cICA) [14] is adopted to systematically form a constrained optimization problem minimizing a new objective function subject to the additional constraints that the extracted ICs are the closest to the corresponding reference signals. An efficient adaptive algorithm to solve this constrained optimization problem is proposed using a Newton-like learning technique. We refer our approach as the *ICA with Reference* (*ICA-R*). The motivation of the ICA-R is to perform both the separation of independent sources and the selection of the desired sources, simultaneously, in a single stage. There are numerous advantages of the ICA-R technique over the previous approaches: it produces only the desired independent sources and facilitates subsequent applications; the computation time and storage requirements are reduced; and the incorporation of the reference signals improves the quality and accuracy of the separation of the interested components.

Let us denote the time varying observed signal by $\mathbf{x}(t) = (x_1(t), x_2(t), \cdots x_n(t))^{\mathrm{T}}$ and the source signal consisting of independent components (ICs) by $\mathbf{c}(t) = (c_1(t), c_2(t), \cdots c_m(t))^{\mathrm{T}}$. In the linear ICA, the signal $\mathbf{x}(t)$ is assumed to be an instantaneous linear mixture of ICs or independent sources c_i, $i = 1, 2, \cdots m$, and therefore:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{c}(t) \tag{1}$$

where the matrix $\mathbf{A}$ of size $n \times m$ represents linear memoryless mixing channels. One often assumes that $n = m$, which case is referred to as *complete* ICA; we hold this assumption in this manuscript for simplicity, which can be relaxed without losing generality. The time index, t, is omitted in the sequel to simplify the notations.

The problem of the ICA-R, addressed in this manuscript, is to learn the demixing matrix, say $\mathbf{W}$, taking a

neural network approach, to estimate the desired components mixed in the input when corresponding reference signals, say $\mathbf{r} = (r_1, r_2, \cdots r_l)^{\mathrm{T}}$, are available; $l\,(< m)$ denotes the number of desired sources to be extracted. By adopting a single-layer feedforward neural network, the desired components recovered at the output, $\mathbf{y} = (y_1, y_2, \ldots y_l)$, are given by

$$\mathbf{y} = \mathbf{W}\mathbf{x} \tag{2}$$

where the weight matrix, $\mathbf{W} = [\mathbf{w}_1\ \mathbf{w}_2\ \cdots\ \mathbf{w}_l]^{\mathrm{T}}$in which $\mathbf{w}_i = (w_{i1}, w_{i2}, \cdots w_{in})^{\mathrm{T}}$corresponds to the weight vector connected to the i th output neuron. In a single-source extraction, we denote the output $y = \mathbf{w}^{\mathrm{T}}\mathbf{x}$ where $\mathbf{w}$ is a weight vector, and the reference signal as r.

The next section summarizes the earlier methods used in extracting a subset of sources from their linear mixtures. Section 3 introduces the technique of ICA-R and derives the Newton-like learning algorithm, section 4 provides analyses on the convergence of the algorithm and discussions on the effect of the relevant parameters, and section 5 demonstrates the present technique with experiments using synthetic signals and real fMRI data. The final section provides discussion and conclusion.

2 Previous Methods

In this section, we discuss several earlier approaches to extract a subset of sources from their mixtures and highlight their drawbacks.

2.1 Second-Order Approach

Second-order approaches use second-order distance criteria, such as MMSE or Maximum Correlation (MC), to obtain the sources close to the references from their mixtures [3]. As an example, by minimizing the Mean Square Error (MSE) between the estimated output $\mathbf{y}$ and the reference signal $\mathbf{r}$, we have the optimum weight $\mathbf{W}^* = [\mathbf{w}_1^* \ \mathbf{w}_2^* \ \cdots \ \mathbf{w}_l^*]^{\mathrm{T}}$ with each element $\mathbf{w}_i^*$, $i = 1, 2, \cdots l$, given by

$$\mathbf{w}_i^* = \sum_{\mathbf{xx}}^{-1} E\{r_i \mathbf{x}\} \tag{3}$$

where $\sum_{\mathbf{xx}}$ is the covariance matrix of the input mixtures $\mathbf{x}$. $E\{\cdot\}$ denotes the temporal expectation. The corresponding output $\mathbf{y}^* = \mathbf{W}^*\mathbf{x}$ provides an approximation close to the reference signal in second-order statistics. However, this approach is insufficient to recover independent sources from the observations, in the higher-order statistical sense. It can be easily shown that as long as more than one IC have non-zero correlations with the reference signal, the extraction of statistically independent sources cannot be achieved by the second-order techniques even if a reference signal is available.

2.2 Classical ICA with Post-Selection

A general equation for learning the demixing matrix, $\mathbf{W}$, using a natural gradient descent method is given by [4, 5, 6, 15]:

$$\Delta\mathbf{W} = \eta \left(\mathbf{I} + \boldsymbol{\phi}(\mathbf{y})\mathbf{y}^{\mathrm{T}}\right) \mathbf{W} \tag{4}$$

where $\eta(< 1.0)$ is the positive learning parameter and $\boldsymbol{\phi}(\mathbf{y}) = (\phi_1(y_1), \phi_2(y_2), \cdots \phi_l(y_l))^{\mathrm{T}}$ in which $\phi_i(y_i)$ is a nonlinear function that depends on the probability density function (pdf) of the component y_i. Many classical ICA

learning algorithms [4, 5, 6] fall into the framework of single-layer feedforward neural networks learning with Eq. (4) [15, 16, 17], but differ only from the formulation of the nonlinear function ϕ_i [15].

Because the classical ICA algorithms produce an output with the same number of components as the input mixtures, a post-process of selecting the desired sources corresponding to the reference signals is required afterwards. Moreover, this two-stage approach is ineffective since it performs redundant work for estimating the uninteresting components. The system performance becomes even worse when high-dimensional input mixtures are involved or only a few output components are interested.

2.3 Nonlinear PCA

Nonlinear PCA algorithms have been derived by introducing nonlinearities into the known objective functions used in the PCA [7, 18]. When the input data are preprocessed by a whitening process, $\mathbf{V}$, the optimization of the nonlinear PCA criterion enables the estimation of the ICs mixed in the inputs. A nonlinear PCA subspace algorithm [7] gives the updating rule for the orthogonal separating matrix, $\widehat{\mathbf{W}}$, and hence provides the demixing matrix: $\mathbf{W} = \widehat{\mathbf{W}}\mathbf{V}$.

The nonlinear PCA rule can be realized by making simple modifications to the standard PCA algorithm and can achieve a dimension reduction in the projection of independent sources. However, the reference signals are not used in this algorithm, so the sources extracted are arbitrary and cannot be specified. Moreover, it should also be noted that the pre-whitening stage of this approach may cause poor separation of sources when the mixing matrices are ill-conditioned or the sources are weak [19]. Therefore, the separation of the desired signals in a single stage from the raw input data is always preferred.

2.4 FastICA

FastICA was recently proposed as a technique to efficiently perform the ICA for one-unit or multi-unit output [8]. The negentropy [20] was utilized as the natural information-theoretic contrast function. Approximations to negentropy [21] were proposed in the fastICA to overcome the statistical limitations of using high-order cumulants.

With the fixed-point method [22], the fastICA was able to extract a subspace of ICs efficiently [8]. However, after each iteration, the demixing matrix needs to be altered by normalizing each weight vector and transforming the geometry of weight elements either in a sequential or parallel way to produce uncorrelated output components for preventing different neurons from converging to the same optimum [8]. This may need extra computations and, in some situations, affect the search of the original sources. The fastICA cannot theoretically obtain particular desired independent sources other than those having the maximum negentropy among the sources. Furthermore, this algorithm may also arbitrarily converge to different local maxima from time to time because the local convergence depends on a number of factors, such as the initial weight vector and the learning rates [23].

3 ICA with Reference (ICA-R)

The ICA-R, a variation to the classical ICA paradigm, is designed to extract a desired subset of ICs and discard the rest of components as irrelevant signals when a set of reference signals representing rough templates of the desired ICs is available. The goal here is to obtain a learning algorithm that satisfies two conditions, simultaneously: (1) the estimated output is a subset of ICs mixed in the input data and (2) the extracted ICs are the closest to the corresponding reference signals, according to some distance criteria.

In the classical ICA, mutual information is adopted as the ICA contrast function by Amari et al. [5] or equivalently as information entropy by Bell and Sejnowski [4]. By minimizing the mutual information or its equivalence, the learning algorithms produce an output with mutually independent components. If the output represents the full-space of original ICs with the same dimension, the classical ICA algorithms [4, 5, 6] are able to recover the original sources mixed in the observed inputs. However, when estimating an output with fewer components than the original ICs, these algorithms may generate some or all of $l(< m)$ output components y_i, $i = 1, 2, \cdots l$, as mixtures of the original ICs, c_k, $k = 1, 2, \cdots m$, though the output components y_i are still mutually independent. It has been proven by Cao and Liu [24] that such systems are l-row decomposable (independent) but their output may not be the original ICs for any $l < m$ when the mixing matrix is of full column rank. Therefore, the sole criterion of statistical independence is inadequate for extracting a subset of original sources; a different objective function is required to resolve such a problem.

The negentropy of a signal y is given by:

$$\mathcal{J}(y) = H(y_{\text{Gaus}}) - H(y) \tag{5}$$

where y_{Gaus} is a Gaussian random variable having the same variance as the signal y and $H(\cdot)$ denotes the entropy of the signal. The negentropy is always positive as a Gaussianly distributed signal has the maximum entropy among all signals [25], and can be interpreted as a measure of non-Gaussianity [20]; large negentropy indicates that the probability distribution is far away from the Gaussian distribution. As it was originally motivated by the information-theoretic index used for exploratory projection pursuit [26], maximizing the negentropy of individual component finds the direction to the original ICs that are non-Gaussian signals mixed in the observed inputs

[27]. Extended to multiple outputs, maximizing their marginal negentropies projects the input data onto a low-dimensional subspace and searches for the structure of non-Gaussianity in the projection [26]. It has been used to achieve the separation of the ICs from their mixtures [8, 26, 27] because the independent sources considered in the ICA usually are non-Gaussian. Then, the first goal of ICA-R becomes the maximization of the sum of marginal negentropies to find the ICs having mostly non-Gaussian distributions and hence, the ICA-R contrast function, $\mathcal{C}(\mathbf{y})$, is defined as

$$\mathcal{C}(\mathbf{y}) = -\sum_{i=1}^{l} \mathcal{J}(y_i) \tag{6}$$

where $\mathcal{J}(y_i)$ is estimated by a flexible and reliable approximation as given in [21]:

$$\mathcal{J}(y_i) \approx \rho \left(E\{f_i(y_i)\} - E\{f_i(\nu)\}\right)^2 \tag{7}$$

where ρ is a positive constant, $f_i(\cdot)$ is a non-quadratic function and ν is a Gaussian variable having zero mean and unit variance. As the optimization of Eq. (6) achieves a non-Gaussian signal for each neuron individually, it may result in different neurons estimating the same independent source. In order to produce different ICs at the output, the additional constraint of uncorrelation among estimated ICs is introduced. The uncorrelation is defined as a pairwise second-order statistical measure between any two different output components, which can be expressed in equality constraints:

$$h_{ij}(y_i, y_j) = \left(E\{y_i y_j\}\right)^2 = 0, \, \forall \, i, j = 1, 2, \cdots l; \, i \neq j. \tag{8}$$

The closeness between the estimated output y_i and the corresponding reference r_i is measured by some norm, denoted by $\varepsilon_i(y_i, r_i)$. The minimum value of $\varepsilon_i(y_i, r_i)$ for all outputs indicates that the estimated output y_i produces the desired IC closest to the corresponding reference signal r_i. An output corresponding to a local optimal solution may give rise to other $m-1$ ICs, c_k, $k = 1, 2, \cdots m; k \neq i$. If we assume that the desired IC is the one and only one closest to the reference r_i,

$$\varepsilon_i(y_i^*, r_i) < \varepsilon_i(y_i^\circ, r_i) \tag{9}$$

where y_i^* denotes the output producing the desired IC which is closest to r_i, and y_i° denotes a local optimum solution having the next minimum value of the closeness measure. Thus, a constraint is defined for the desired output component with the closeness measure to r_i less than or equal to a threshold parameter ξ_i: $g_i(y_i) =$

$\varepsilon_i(y_i, r_i) - \xi_i \leq 0$ only when $y_i = y_i^*$ and not with any other $m-1$ local optimal solutions if we choose the threshold ξ_i in the scalar range Υ_i given by

$$\Upsilon_i = [\varepsilon_i(y_i^*, r_i),\ \varepsilon_i(y_i^\circ, r_i))\,. \tag{10}$$

Incorporating the uncorrelation and the restriction of the closeness measures as the feasible equality and inequality constraints into the contrast function in Eq. (6), the problem of ICA-R can be modeled in the cICA framework as a constrained optimization problem [14]:

$$\begin{aligned} &\text{minimize} \quad \mathcal{C}(\mathbf{y}) \\ &\text{subject to} \quad \mathbf{g}(\mathbf{y}) \leq \mathbf{0} \ \text{ and } \ \mathbf{h}(\mathbf{y}) = \mathbf{0} \end{aligned} \tag{11}$$

where the inequality constraint term, $\mathbf{g}(\mathbf{y}) = (g_1(y_1), g_2(y_2), \cdots g_l(y_l))^{\mathrm{T}}$ with $g_i(y_i) = \varepsilon_i(y_i, r_i) - \xi_i$, and the equality constraint term, $\mathbf{h}(\mathbf{y}) = (h_{11}(y_1), h_{12}(y_1, y_2), \cdots h_{1l}(y_1, y_l), h_{21}(y_2, y_1) \cdots h_{ll}(y_l))^{\mathrm{T}}$ denote a vector of l^2 functions where the pairwise correlations, $h_{ij}(y_i, y_j)$, $i \neq j$, as in Eq. (8) and elements $h_{ii}(y_i) = (E\{y_i^2\} - 1)^2$, $\forall i = 1, 2, \cdots l$ are added to restrict each output have unit variance.

To simplify the optimization problem, the inequality constraints are transformed into equality constraints, $\hat{g}_i(y_i) = g_i(y_i) + z_i^2 = 0, \forall i = 1, 2, \cdots l$, by introducing a vector of slack variables, $\mathbf{z} = (z_1, z_2, \cdots z_l)^{\mathrm{T}}$. The Lagrange multipliers method [28] is adopted to search for the optimal solution in the ICA-R; the corresponding augmented Lagrangian function $\mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{z})$ is given by

$$\mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{z}) = \mathcal{C}(\mathbf{y}) + \boldsymbol{\mu}^{\mathrm{T}} \hat{\mathbf{g}}(\mathbf{y}) + \frac{1}{2}\gamma \left\| \hat{\mathbf{g}}(\mathbf{y}) \right\|^2 + \boldsymbol{\lambda}^{\mathrm{T}} \mathbf{h}(\mathbf{y}) + \frac{1}{2}\gamma \left\| \mathbf{h}(\mathbf{y}) \right\|^2 \tag{12}$$

where $\boldsymbol{\mu} = (\mu_1, \mu_2, \cdots \mu_l)^{\mathrm{T}}$ and the concatenated vector $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_1^{\mathrm{T}}, \boldsymbol{\lambda}_2^{\mathrm{T}}, \cdots \boldsymbol{\lambda}_l^{\mathrm{T}})^{\mathrm{T}}$ in which $\boldsymbol{\lambda}_i = (\lambda_{i1}, \lambda_{i2}, \cdots \lambda_{il})^{\mathrm{T}}$, $\forall i = 1, 2, \cdots l$, are the vectors of positive Lagrange multipliers for inequality constraints and equality constraints, respectively, elements μ_i and λ_{ij} are related to the corresponding constraint term $g_i(y_i)$ and $h_{ij}(y_i, y_j)$, $\forall i, j = 1, 2, \cdots l$, $\gamma\ (> 0)$ is the penalty parameter, and $||\cdot||$ denotes the Euclidean norm. The quadratic penalty term $\frac{1}{2}\gamma||\cdot||^2$ ensures that the optimization problem is held at the condition of local convexity assumption [28].

Because the minimization of $\mathcal{L}$ with respect to $\mathbf{z}$ can be carried out explicitly for fixed $\mathbf{W}$, the function of Eq.

(12) is first minimized with respect to z_i, $\forall i = 1, 2, \cdots l$:

$$\min_{z_i} \mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{z}) = \min_{z_i^2 \geq 0} \left\{ \mu_i \left(g_i(y_i) + z_i^2 \right) + \frac{1}{2} \gamma \left\| g_i(y_i) + z_i^2 \right\|^2 \right\} \tag{13}$$

At the minimal point, the first derivative $\nabla_{z_i} \mathcal{L} = 0$ and $z_i^2 \geq 0$; if z_i^* denotes the optimal value of z_i, $(z_i^*)^2 = \max \left\{ 0, -\left(\frac{\mu_i}{\gamma} + g_i(y_i) \right) \right\}$. By substituting $(z_i^*)^2$ in Eq. (12), the ICA-R objective function is given by

$$\mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \mathcal{C}(\mathbf{y}) + \mathcal{G}(\mathbf{y}, \boldsymbol{\mu}) + \mathcal{H}(\mathbf{y}, \boldsymbol{\lambda}) \tag{14}$$

where $\mathcal{G}(\mathbf{y}, \boldsymbol{\mu}) = \frac{1}{2\gamma} \sum_{i=1}^{l} \left\{ \max^2 \{0, \mu_i + \gamma g_i(y_i)\} - \mu_i^2 \right\}$ denotes the term for inequality constraints of restricting the closeness measures and $\mathcal{H}(\mathbf{y}, \boldsymbol{\lambda}) = \boldsymbol{\lambda}^{\mathrm{T}} \mathbf{h}(\mathbf{y}) + \frac{1}{2} \gamma ||\mathbf{h}(\mathbf{y})||^2$ corresponds to the term for equality constraints of uncorrelation. In order to have a stable and efficient learning rule, we derive a Newton-like learning algorithm for the demixing matrix, $\mathbf{W}$, in its vector form, as

$$\Delta \mathrm{Vec}\left(\mathbf{W}^{\mathrm{T}}\right) = -\eta \left(\nabla^2_{\mathrm{Vec}(\mathbf{W}^{\mathrm{T}})} \mathcal{L} \right)^{-1} \nabla_{\mathrm{Vec}(\mathbf{W}^{\mathrm{T}})} \mathcal{L} \tag{15}$$

where the learning rate η is usually equal to one, but may be decreased gradually to ensure the stable convergence, $\mathrm{Vec}(\cdot)$ is an operator on a matrix, which cascades the columns of the matrix from left to right and forms a column vector; $\nabla_{\mathrm{Vec}(\mathbf{W}^{\mathrm{T}})} \mathcal{L} = \dfrac{\partial \mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda})}{\partial \mathrm{Vec}\left(\mathbf{W}^{\mathrm{T}}\right)}$ and $\nabla^2_{\mathrm{Vec}(\mathbf{W}^{\mathrm{T}})} \mathcal{L} = \dfrac{\partial^2 \mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda})}{\partial \mathrm{Vec}^2\left(\mathbf{W}^{\mathrm{T}}\right)}$ are the corresponding gradient vector and Hessian matrix of the objective function in Eq. (14), with respect to $\mathrm{Vec}\left(\mathbf{W}^{\mathrm{T}}\right)$, respectively.

The gradient of the objective function $\mathcal{L}$ is given by:

$$\nabla_{\mathbf{W}} \mathcal{L}(\mathbf{W}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = E\{\nabla_{\mathbf{y}} \mathcal{C}(\mathbf{y})\, \mathbf{x}^{\mathrm{T}}\} + E\{\nabla_{\mathbf{y}} \mathcal{G}(\mathbf{y}, \boldsymbol{\mu})\, \mathbf{x}^{\mathrm{T}}\} + 2 \nabla_{\sum_{\mathbf{yy}}} \mathcal{H}(\mathbf{y}, \boldsymbol{\lambda})\, E\{\mathbf{y}\mathbf{x}^{\mathrm{T}}\} \tag{16}$$

where the vector $\nabla_{\mathbf{y}} \mathcal{C}(\mathbf{y}) = (\mathcal{C}'_{y_1}(\mathbf{y}), \mathcal{C}'_{y_2}(\mathbf{y}), \cdots \mathcal{C}'_{y_l}(\mathbf{y}))^{\mathrm{T}}$ with $\mathcal{C}'_{y_i}(\mathbf{y}) = -\hat{\rho}_i\, f'_{y_i}(y_i)$ in which $\hat{\rho}_i = 2\rho(E\{f_i(y_i)\} - E\{f_i(\nu)\})$, $f_i(\cdot)$ is the non-quadratic function in Eq. (7) to approximate the negentropy and $f'_{y_i}(y_i)$ is the first derivative of $f_i(y_i)$ with respect to y_i. The vector $\nabla_{\mathbf{y}} \mathcal{G}(\mathbf{y}, \boldsymbol{\mu}) = (\mathcal{G}'_{y_1}(\mathbf{y}, \boldsymbol{\mu}), \mathcal{G}'_{y_2}(\mathbf{y}, \boldsymbol{\mu}), \cdots \mathcal{G}'_{y_l}(\mathbf{y}, \boldsymbol{\mu}))^{\mathrm{T}}$ has $\mathcal{G}'_{y_i}(\mathbf{y}, \boldsymbol{\mu}) = \mu_i\, g'_{y_i}(y_i)$, and the matrix $\nabla_{\sum_{\mathbf{yy}}} \mathcal{H} = \left[\nabla_{\sum_{y_1 \mathbf{y}}} \mathcal{H} \;\; \nabla_{\sum_{y_2 \mathbf{y}}} \mathcal{H} \;\; \cdots \;\; \nabla_{\sum_{y_l \mathbf{y}}} \mathcal{H} \right]^{\mathrm{T}}$ is the gradient of $\mathcal{H}(\mathbf{y}, \boldsymbol{\lambda})$ with respect to the covariance matrix of output $\mathbf{y}$, $\sum_{\mathbf{yy}}$; the i th row $\nabla_{\sum_{y_i \mathbf{y}}} \mathcal{H} = \left(\mathcal{H}'_{\sum_{y_i y_1}}, \mathcal{H}'_{\sum_{y_i y_2}}, \cdots \mathcal{H}'_{\sum_{y_i y_l}} \right)^{\mathrm{T}}$ where $\mathcal{H}'_{\sum_{y_i y_j}} = 2\lambda_{ij}(E\{y_i y_j\} - \delta_{ij})$ in which $\sum_{y_i y_j} = E\{y_i y_j\}$ and δ_{ij} is zero when $i \neq j$, and equals to one, otherwise. By transforming the demixing matrix, $\mathbf{W}$, and its gradient in Eq. (16) into vectors, the Hessian matrix

is obtained by partially differentiating the gradient with respect to the $\mathrm{Vec}\left(\mathbf{W}^{\mathrm{T}}\right)$. To simplify the matrix inversion and keep the stability of the Newton-like method, we approximate the Hessian matrix as

$$\nabla^2_{\mathrm{Vec}(\mathbf{W}^{\mathrm{T}})}\mathcal{L}(\mathbf{W},\boldsymbol{\mu},\boldsymbol{\lambda}) \approx \mathbf{D}\otimes\sum\nolimits_{\mathbf{xx}} \tag{17}$$

where $\mathbf{D}$ is a diagonal matrix in which the diagonal is a vector $\mathbf{d}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda}) = (d_1(y_1,\mu_1,\lambda_{11}), d_2(y_2,\mu_2,\lambda_{22}), \cdots d_l(y_l,\mu_l,\lambda_{ll}))^{\mathrm{T}}$ with $d_i(y_i,\mu_i,\lambda_{ii}) = E\{\mu_i\, g''_{y_i}(y_i)\} + 8\lambda_{ii} - E\{\hat{\rho}_i\, f''_{y_i}(y_i)\}$ where $g''_{y_i}(y_i)$ and $f''_{y_i}(y_i)$ are the second derivatives of $g_i(y_i)$ and $f_i(y_i)$ with respect to y_i, $\sum_{\mathbf{xx}}$ is the covariance matrix of input $\mathbf{x}$, and $\otimes$ denotes the Kronecker product of two matrices.

Applying the relational properties: $(\mathbf{A}\otimes\mathbf{B})^{-1} = \mathbf{A}^{-1}\otimes\mathbf{B}^{-1}$ and $\mathrm{Vec}(\mathbf{ABC}) = (\mathbf{C}^{\mathrm{T}}\otimes\mathbf{A})\mathrm{Vec}(\mathbf{B})$, the Newton-like learning rule in Eq. (15) is simplified to

$$\Delta\mathrm{Vec}\left(\mathbf{W}^{\mathrm{T}}\right) = -\eta\mathrm{Vec}\left(\textstyle\sum_{\mathbf{xx}}^{-1}\ \nabla_{\mathbf{W}^{\mathrm{T}}}\mathcal{L}(\mathbf{W},\boldsymbol{\mu},\boldsymbol{\lambda})\ \mathbf{D}^{-1}\right). \tag{18}$$

By transforming the vector form back to the matrix, the learning for $\mathbf{W}$ is obtained as

$$\Delta\mathbf{W} = -\eta\left(\boldsymbol{\Phi}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda}) + \boldsymbol{\Psi}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda}) + \boldsymbol{\Omega}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda})\right)\textstyle\sum_{\mathbf{xx}}^{-1} \tag{19}$$

where the matrices, $\boldsymbol{\Phi}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda}) = [\boldsymbol{\phi}_1(y_1,\mu_1,\lambda_{11})\ \boldsymbol{\phi}_2(y_2,\mu_2,\lambda_{22})\ \cdots\ \boldsymbol{\phi}_l(y_l,\mu_l,\lambda_{ll})]^{\mathrm{T}}$ with $\boldsymbol{\phi}_i(y_i,\mu_i,\lambda_{ii}) = E\{\mathcal{C}'_{y_i}(\mathbf{y})\mathbf{x}\}/d_i(y_i,\mu_i,\lambda_{ii})$, $\boldsymbol{\Psi}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda}) = [\boldsymbol{\psi}_1(y_1,\mu_1,\lambda_{11})\ \boldsymbol{\psi}_2(y_2,\mu_2,\lambda_{22})\ \cdots\boldsymbol{\psi}_l(y_l,\mu_l,\lambda_{ll})]^{\mathrm{T}}$ with $\boldsymbol{\psi}_i(y_i,\mu_i,\lambda_{ii}) = E\{\mathcal{G}'_{y_i}(\mathbf{y},\boldsymbol{\mu})\mathbf{x}\}/d_i(y_i,\mu_i,\lambda_{ii})$, and $\boldsymbol{\Omega}(\mathbf{y},\boldsymbol{\mu},\boldsymbol{\lambda}) = [\boldsymbol{\omega}_1(\mathbf{y},\mu_1,\boldsymbol{\lambda}_1)\,\boldsymbol{\omega}_2(\mathbf{y},\mu_2,\boldsymbol{\lambda}_2)\ \cdots\ \boldsymbol{\omega}_l(\mathbf{y},\mu_l,\boldsymbol{\lambda}_l)]^{\mathrm{T}}$ with $\boldsymbol{\omega}_i(\mathbf{y},\mu_i,\boldsymbol{\lambda}_i) = 2\,E\{\mathbf{x}\mathbf{y}^{\mathrm{T}}\}\,\nabla_{\sum_{y_i\mathbf{y}}}\mathcal{H}(\mathbf{y},\boldsymbol{\lambda})/d_i(y_i,\mu_i,\lambda_{ii})$. The expectation in the equations is estimated using all the samples of the input $\mathbf{x}$ and the output $\mathbf{y}$. The Lagrange multipliers in $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ are initialized to zeros and updated using the gradient ascent equations:

$$\Delta\boldsymbol{\mu} = \max\{-\boldsymbol{\mu}, \gamma\,\mathbf{g}(\mathbf{y})\}, \tag{20}$$

$$\Delta\boldsymbol{\lambda} = \gamma\,\mathbf{h}(\mathbf{y}) \tag{21}$$

where γ is the learning rate.

For the special case of single-unit extraction, the uncorrelation terms in the objective function and the learning algorithm disappear. Therefore, the Newton-like learning rule simplifies to

$$\Delta\mathbf{w} = -\eta(\boldsymbol{\phi}(y,\mu,\lambda) + \boldsymbol{\psi}(y,\mu,\lambda) + \boldsymbol{\omega}(y,\mu,\lambda))\textstyle\sum_{\mathbf{xx}}^{-1} \tag{22}$$

where $\boldsymbol{\phi}(y,\mu,\lambda) = E\{\mathcal{C}'(y)\,\mathbf{x}\}/d(y,\mu,\lambda)$, $\boldsymbol{\psi}(y,\mu,\lambda) = E\{\mathcal{G}'(y,\mu)\,\mathbf{x}\}/d(y,\mu,\lambda)$, $\boldsymbol{\omega}(y,\mu,\lambda) = 4\lambda(E\{y^2\} - 1)E\{y\,\mathbf{x}\}/d(y,\mu,\lambda)$, and $d(y,\mu,\lambda) = E\{\mu g''(y)\} + 8\lambda - E\{\hat{\rho} f''(y)\}$.

4 Analyses of Convergence Stability and Parameter Selection

This section presents an analysis on the minimum points and the convergence stability of the algorithm. The selection of relevant parameters and their effects to the convergence are discussed. Theoretically, with the Newton-like learning presented above, the neural network is able to achieve the minimum points and produce the optimum output $\mathbf{y}^*$ to extract the desired ICs, defined by Kuhn-Tucker (KT) triple $(\mathbf{W}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ that satisfies the first-order conditions: $\nabla_{\mathbf{W}}\mathcal{L}(\mathbf{W}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) = \mathbf{0}$; $\mathbf{h}(\mathbf{y}^*) = \mathbf{0}$; $\mathbf{g}(\mathbf{y}^*) \leq \mathbf{0}$; $\boldsymbol{\lambda}^* > \mathbf{0}$, $\boldsymbol{\mu}^* \geq \mathbf{0}$ and $\boldsymbol{\mu}^{*\mathrm{T}}\mathbf{g}(\mathbf{y}^*) = \mathbf{0}$. The '$*$' denotes the optimal value of the parameters.

However, the convergence of the learning algorithm also depends on the selected thresholds $\boldsymbol{\xi}$ defined in the inequality constraint term, $\mathbf{g}(\mathbf{y})$. In single-source extraction, the desired IC is determined by the value of the single threshold, say ξ. Any IC, c_k, other than the desired sources, having the closeness measure $\varepsilon(c_k, r) \leq \xi$ is treated as the local optima for the extraction. By selecting a suitable ξ having the range given as in Eq. (10), the one and only one desired IC is determined as the optimum output; the algorithm converges to the global minimum of the objective function. However, if ξ is beyond the upper bound of the range, that output component may have more than one convergent point. If ξ is too small, the component y may be unable to produce any desired ICs because the corresponding constraint $g(y) \gg 0$ causes learning of that neuron to become unpredictable. In practice, the algorithm better uses a small ξ initially to avoid having too many local minimum points and then gradually increases so that the algorithm converges at the global minimum point corresponding to the one and only one desired IC. This discussion can be extended to the cases of multiple neurons ($l > 1$) to select and adjust the threshold for each neuron, individually. If all reference signals are different from one another and all the thresholds are adjusted properly, the algorithm achieves the global minimum rendering all the desired ICs.

Let us examine the stability of the convergence of the algorithm. Suppose that the network is in a local minimum and perturbed by a small vector $\boldsymbol{\epsilon}$ to $\mathbf{W}^*$ in vector form $\mathrm{Vec}\left(\mathbf{W}^{*\mathrm{T}}\right)$. By a truncated Taylor series expansion,

$$\begin{aligned}\mathcal{L}(\mathrm{Vec}\left(\mathbf{W}^{*\mathrm{T}}\right) + \boldsymbol{\epsilon}, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) \quad &\approx \quad \mathcal{L}(\mathbf{W}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) + \boldsymbol{\epsilon}^{\mathrm{T}}\nabla_{\mathrm{Vec}(\mathbf{W}^{*\mathrm{T}})}\mathcal{L}(\mathbf{W}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) \\ &\quad + \tfrac{1}{2}\boldsymbol{\epsilon}^{\mathrm{T}}\nabla^2_{\mathrm{Vec}(\mathbf{W}^{*\mathrm{T}})}\mathcal{L}(\mathbf{W}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)\,\boldsymbol{\epsilon}.\end{aligned} \tag{23}$$

The second term is equal to 0 at the local minimum. For the network to converge at the minimum point, the third term should always be positive for any small $\boldsymbol{\epsilon}$. In other words, the Hessian matrix $\nabla^2_{\mathrm{Vec}(\mathbf{W}^{*\mathrm{T}})}\mathcal{L}$ should be positive definite, which is also known as the second-order condition. For simplicity, let us consider the approximated $\nabla^2_{\mathrm{Vec}(\mathbf{W}^{*\mathrm{T}})}\mathcal{L}$ in Eq. (17) although our examination is valid for the original Hessian as well. $\nabla^2_{\mathrm{Vec}(\mathbf{W}^{*\mathrm{T}})}\mathcal{L}$ is always positive definite when the input covariance matrix $\sum_{\mathbf{xx}}$ is non-singular and all elements, $d(y_i^*, \mu_i^*, \lambda_{ii}^*)$, $\forall i = 1, 2, \cdots l$, along the diagonal of the diagonal matrix $\mathbf{D}$ are positive. The former condition is true in most cases when a large number of sample points of signals is available. Even when it is singular or near-singular if the input consists of a small number of samples, $\sum_{\mathbf{xx}}$ can be transformed into a non-singular matrix by applying a whitening process, e.g. PCA.

To examine the latter condition, it is important to select a suitable distance criterion as the closeness measure $\varepsilon_i(y_i, r_i)$ between the output y_i and the reference signal r_i, which is the key function in inequality constraints of $g_i(y_i)$. The proper choice of closeness makes the algorithm stable and robust and may also facilitate the determination of the suitable threshold ξ_i. A common and simple measure of closeness is the MSE: $\varepsilon_i(y_i, r_i) = E\{(y_i - r_i)^2\}$. This measure requires both y_i and r_i normalized to have same means and variances. Alternatively, correlation can also be used as a closeness measure such that $\varepsilon_i(y_i, r_i) = 1/\left(E\{y_i\, r_i\}\right)^2$. Both the output and reference signals need to be normalized so that the value of the correlation is bounded. The corresponding first and second-order derivatives of $g_i(y_i)$ are

$$g'_{\mathrm{cor}}(y_i) \;=\; -2\frac{E\{r_i\}}{\left(E\{y_i\, r_i\}\right)^3} \;\text{ and }\; g''_{\mathrm{cor}}(y_i) = 6\frac{\left(E\{r_i\}\right)^2}{\left(E\{y_i\, r_i\}\right)^4}. \tag{24}$$

The selection of the closeness measure may be different from one neuron to another, depending on what form the individual reference signals are available. Using either MSE or correlation as the function of $\varepsilon_i(y_i, r_i)$, the term $\mu_i^*\, g''_{y_i^*}(y_i^*)$ is always positive since the Lagrange multipliers, μ_is, hold the positivity property ensured by their learning rules given in Eq. (20).

The Hessian matrix $\nabla^2_{\mathrm{Vec}(\mathbf{W}^{\mathrm{T}})}\mathcal{H}(\mathbf{y}, \boldsymbol{\lambda})$ is derived by differentiating the gradient of the uncorrelation constraints $\mathcal{H}(\mathbf{y}, \boldsymbol{\lambda})$ in Eq. (16), $\mathrm{Vec}\left(2\nabla_{\sum_{\mathbf{yy}}}\mathcal{H}(\mathbf{y}, \boldsymbol{\lambda})\, E\{\mathbf{y}\mathbf{x}^{\mathrm{T}}\}\right)$, by the $\mathrm{Vec}\left(\mathbf{W}^{\mathrm{T}}\right)$. At the optima, the Hessian matrix is

simplified by the block diagonal matrix:

$$\nabla^2_{\mathrm{Vec}(\mathbf{W}^{*\mathrm{T}})}\mathcal{H}(\mathbf{y}^*, \boldsymbol{\lambda}^*) = \mathbf{D}^*_{\mathcal{H}} \otimes \sum_{\mathbf{xx}} \tag{25}$$

where $\mathbf{D}^*_{\mathcal{H}}$ is the diagonal matrix having the diagonal elements $d^*_{\mathcal{H}}(y^*_i, \lambda^*_{ii}) = 4\lambda^*_{ii}(3E\{y^{*2}_i\} - 1), \forall i = 1, 2, \cdots l$. Because the variance of y^*_i is equal to one at the optimum point, $d^*_{\mathcal{H}}(y^*_i, \lambda^*_{ii}) = 8\lambda^*_{ii}$ and is always positive as the Lagrange multipliers, λ_{ii}s, hold the positivity property ensured by their learning rules given in Eq. (21). In order to keep the learning stable, we approximate the term in $d_i(y_i, \mu_i, \lambda_{ii})$ corresponding to the uncorrelation constraints by $8\lambda_{ii}$.

We propose the following two practical functions for the nonlinear function, $f(y)$, used in Eq. (7) to approximate the negentropy of the super-Gaussian and sub-Gaussian signals, respectively:

$$f_{\mathrm{sup}}(y) = \frac{1}{a}\log\cosh(ay) - \frac{a}{2}y^2, \tag{26}$$

$$f_{\mathrm{sub}}(y) = \frac{b}{4}y^4 \tag{27}$$

where $a, b \in \mathbf{R}^+$. The density type of either super-Gaussian or sub-Gaussian is determined by the sign of the normalized kurtosis measure $\kappa_4(y) = \frac{E\{y^4\}}{(E\{y^2\})^2} - 3$. Then

$$f'_{\mathrm{sup}}(y) = \tanh(ay) - ay, \quad f''_{\mathrm{sup}}(y) = a(\mathrm{sech}^2(ay) - 1), \tag{28}$$

$$f'_{\mathrm{sub}}(y) = by^3, \qquad f''_{\mathrm{sub}}(y) = 3by^2, \tag{29}$$

where $f''_{\mathrm{sup}}(y)$ is always non-positive, and $f''_{\mathrm{sub}}(y)$ is always non-negative. It is also easy to verify that the value of $\hat{\rho}^*_i = 2\rho(E\{f_i(y^*_i)\} - E\{f_i(\nu)\})$ at the optimal solution, y^*_i, is always positive for super-Gaussian distributions and negative for sub-Gaussian distributions when the nonlinear functions are chosen as in Eqs. (26) and (27). Therefore, the product $-\hat{\rho}^*_i\, f''_{y_i}(y^*_i)$ has a non-negative value for both super- and sub-Gaussian signals when using the above form of nonlinear functions. That is, the convergence of the learning given by Eqs. (19) - (21) is stable when one uses the above functions: $f_{\mathrm{sup}}(y)$ or $f_{\mathrm{sub}}(y)$ for super- or sub-Gaussian signals, respectively, and the MSE or the correlation as the closeness measure.

The present Newton-like algorithm incorporates equality and inequality constraints and involves Hessian matrices and second-order derivatives in the learning rules, therefore, its computation load corresponding to a single

output component in one iteration is heavier than that in classical ICA algorithms. By comparing the Matlab execution profile, the present algorithm usually uses 1.5 times more floating point operations than the classical ICA per output component per iteratioin. As an example, 50 KFLOPs (Kilo Floating point Operations) are used for the ICA-R, whereas 20 KFLOPs for Lee et al.'s extended infomax ICA algorithm. However, the ICA-R algorithm only extracts the l number of desired signals, which is usually much less than the outputs in the classical ICA that equal to the number of inputs, m. The Newton-like learning rule and the availability of reference signals to select a smaller subset of outputs enable the ICA-R algorithm to stably converge in a less number of iterations. Overall, when the number of desired components is half of the number of inputs, the computational complexity of the ICA-R and classical ICA algorithms is similar, but the ICA-R algorithm converges much faster than the classical ICA when a large number of input signal mixtures involve and only a few of output components are of interests, as in fMRI experiments.

5 Experiments

We demonstrate the technique of the ICA-R by using experiments with a mixture of synthetic signals and real fMRI datasets and compare the accuracy and efficacy of the algorithm with the earlier methods.

5.1 Synthetic Signals

Five zero-mean and unit-variance synthetic independent sources: a Gaussian distributed signal, c_1, two periodic deterministic signals, c_2 and c_3, and two random signals with sub-Gaussian distribution, c_4, and with super-Gaussian distribution, c_5, were randomly mixed to obtain five inputs, which are displayed in figures 1(a) and (b), respectively. The ICA-R, the classical ICA and the second-order approach were used to extract the two deterministic signals, c_2 and c_3, mixed in the inputs. The references for the deterministic signals were simulated by a series of pulses having small widths and the same period as the desired sources. The ICA-R algorithm ran using the correlation as the closeness measure with the corresponding (pulse) reference signals. As expected, the algorithm converged to produce the desired sources; the references and extracted signals are displayed in figures 1(c) and (d).

The accuracies of the recovered ICs compared to the sources were expressed using the signal-to-noise ratio (SNR) in dB given by $\mathrm{SNR} = 10\log_{10}\left(\frac{s^2}{\mathrm{MSE}}\right)$ where s^2 denotes the variance of the source and MSE denotes the mean square error between the original and recovered signals. The SNRs were computed for each output component with all the sources, and the corresponding source was determined by the best SNR value. The performances of the separation of the input signal into the ICs were measured by a performance index (PI) of the permutation error: $\mathrm{PI} = \frac{1}{l}\left(\sum_{i=1}^{l}\mathrm{rPI}_i + \sum_{j^*=1}^{l}\mathrm{cPI}_{j^*}\right)$ where $\mathrm{rPI}_i = \sum_{j=1}^{m}\frac{|p_{ij}|}{\max_k|p_{ik}|} - 1$ in which p_{ij} denotes the (i,j) th element of the permutation matrix $\mathbf{P} = \mathbf{WA}$, and $\mathrm{cPI}_j = \sum_{i=1}^{l}\frac{|p_{ij}|}{\max_k|p_{kj}|} - 1$ in which p_{ij} denotes the elements in $\mathbf{P}$, corresponding to the j th IC in the desired subset of sources. The term rPI_i gives the error of the separation of the output component y_i with respect to the sources and cPI_j measures the degree of the desired IC, c_j, appearing multiple times at the output. PI is zero when the desired subset of ICs is perfectly separated. The measures indicating the quality of the separation and the recovered sources of different techniques are given

in table 1. The high SNRs and low PIs of the present technique indicated superior performance over the other techniques.

The distorted outputs, low SNRs and poor PIs indicate the failure of the second-order method in separating the desired sources. The classical ICA algorithms used the same parameters as the ICA-R to produce an output with five components and then the desired ICs corresponding to the references were selected post hoc. The SNRs of the selected signals and PIs of the algorithms are worse than that of the ICA-R technique as the algorithm attempts to separate all the other uninterested components at the same time. And also, the classical ICA algorithm needed more iterations to converge than the present technique; the performance indices of the two techniques with iterations are shown in figure 2. The comparison of how many MFLOPs (Mega Floating point Operations) used in their learning processes is given in table 2. It shows that the ICA-R algorithm run double faster than the classical ICA in this experiment.

[Figure 1 is to be included here.]

[Figure 2 is to be included here.]

[Table 1 is to be included here.]

[Table 2 is to be included here.]

The nonlinear PCA and fastICA algorithms were also run with the five mixture inputs to extract two ICs; the results are shown in figures 3(a) and (b), respectively. The nonlinear PCA produced different subset of sources close to c_2 and c_4. The fastICA always extracted signals identical to c_2 and c_5 because the deterministic signal, c_2, and the super-Gaussian source, c_5, had the maximum negentropies among all the sources. It is generally impossible to specifically extract the desired signals by using either nonlinear PCA or the fastICA.

[Figure 3 is to be included here.]

5.2 FMRI

Functional Magnetic Resonance Imaging (fMRI) data carry spatio-temporal information of responses of the brain activated by some functional tasks. Because fMRI data are always confounded by physiological signals such as cardiac, respiratory, and blood flow, the electronic noise of the scanners and other environmental artifacts [29, 30, 31, 32], preprocessing to remove the non-task related signals is usually the first step in the analysis of fMRI data. The detection of the brain activation is usually performed by statistically comparing time-series corresponding to each brain voxel with the input stimuli and thereby generating statistical maps. The simplest of this approach is the correlation analysis [12]. In order to correct for multiple comparisons and spatial correlations, the statistical maps are further analyzed using the Gaussian random field (GRF) theory, which approach is referred to as the *statistical parametric mapping* (SPM) [13]. Prior to the simple correlation analysis or the SPM analysis, fMRI data are required to preprocess using appropriate denoising or smoothing filters and correct for artifacts. These preprocessing techniques are still mostly ad hoc and tend to alter the original data.

Recently, the ICA has become an increasingly popular data-driven approach for the analysis of fMRI data because of its capability of separating the components of interest, that are task-related, from other components that are due to interferences and artifacts [30]. Therefore, it is unnecessary to preprocess fMRI data before using the ICA technique for the detection of brain activation. The classical ICA method presumes that the task-related components are independent of non-task related components in spatial-domain and produces all the component maps as the output; post-processing, such as correlation of time responses and stimuli, is necessary to select the components corresponding to the task-related brain activation [33]. However, the spatial independence of the effect of heartbeat, respiration, and blood flow in the brain is questionable because their influence is common to most regions of the brain; it is more appropriate to assume that these signals are mutually independent in time-domain because they have frequencies different from the functional tasks.

The ICA-R technique may be effectively used in most fMRI data analysis because the task-related fMRI responses usually follow the input stimuli. Therefore, in the rest of this section we explore the efficacy of applying ICA-R in the fMRI analysis by using both synthetic and real fMRI datasets. The real fMRI datasets used in this

section were obtained in the experiments performed at 3.0 Tesla Medspec 30/100 scanner at the MRI Center of the Max-Planck-Institute of Cognitive Neuroscience.

The images were analyzed using the techniques of SPM, the classical ICA with post-selection and the ICA-R. The ICA-R assumes the temporal independence among the time courses due to various sources involved in fMRI and automatically extracts only the task-related components by using the input stimuli as the reference signals. In order to find the significantly activated voxels in the SPM analysis, a cluster size threshold of 3 and p-value of 0.05 was used. To find and display voxels contributing significantly to a component map in the ICA techniques, the map values were scaled to z-values; voxels whose absolute z-values greater than 2.5 were considered as the active voxels. The optimal values of the cluster size or significance thresholds were empirically found as there are no techniques to analytically determine these values. Z-statistical scores of the detected significant blobs were superimposed on to the corresponding anatomical slices for visualization.

5.2.1 Synthetic fMRI Data

A two-dimensional 64×64 synthetic functional image consisting of 96 scans was simulated. Four 9×9 square blobs of pixels were selected to be activated, as shown in figure 4(a). The input stimulation was presumed to have twelve cycles, each having two stimulation ON states followed by six OFF states, and each stimulation had a duration of 3s. Box-car time-series were designed for the activated pixels and the inactive pixels had time-series of zero amplitudes. The responses of the activated pixels were generated by convolving the box-car time-series with a gamma HRF [13]. Gaussian random noise was then added to the time-series of both active and inactive pixels. Pixel intensities of an image scan in the synthetic image were given by the values of the time-series at the corresponding time instances. The spatial correlation of the scans was introduced by convolving each functional scan with a Gaussian kernel having FWHM of 3.0 and thereafter the images were properly scaled. If the amplitude of the box-car time-series, represented in the image is s, and the standard deviation of the Gaussian noise is σ, then the SNR in dB is defined as $\mathrm{SNR} = 20\log(\frac{s}{\sigma})$. Figures 4(b) and (c) show the sixth and tenth slices, respectively, of the synthetic image at a SNR of -8.5 dB. The gray contrasts of the figures have been increased 80% for easy visualization of activated regions. The accuracy of detected activation was illustrated by a ROC curve, which

indicates the true-positives vs. the false-positives rates.

[Figure 4 is to be included here.]

The classical ICA approach ran 100 iterations to produce 64 component maps from synthetic images, whereas the present ICA-R technique ran in only 5 iterations to extract the single task-related component. The MFLOPs comparison in table 2 shows the ICA-R algorithm run almost 1000 faster than the classical ICA in this experiment. To find and display pixels significantly activated in the task-related component, the map values were scaled to z-statistical values. The accuracy of detected activation of the four square blobs in above two methods were compared with different *z*-value thresholds: from 1.5 to 3.5 with interval of 0.05. The ROC curves are shown in figures 5. Figures 6(a), (b) and (c) show the detected activations by using the ICA-R, the SPM and the classical ICA approaches at the *z*-value threshold of 2.75 at which all methods showed the lowest error rates. As seen in the figures, the ICA-R technique had the better performance than either the SPM or the classical ICA regardless of the choices of the *z*-value threshold.

[Figure 5 is to be included here.]

[Figure 6 is to be included here.]

5.2.2 Visual Task Data

An 8 Hz alternating checker board pattern with a central fixation point was projected on an LCD system; 5 subjects were asked to fixate on the point of stimulation. A series of 64 FLASH images were acquired at three axial levels at the visual cortex of the brain while the subjects were performing alternatively the stimulation and rest tasks. The parameters of the scanning protocols and more experimental details can be found in Rajapakse et al., 1998.

The brain activation was detected using the ICA with post-selection, the ICA-R, and the SPM technique. The ON-OFF stimulation was used as the reference signal and the correlation was used as the closeness measure for the ICA-R. Unlike the ICA-R, the SPM technique required preprocessing of the fMRI data in order to remove noise, and artifacts due to subjects' head motion; without preprocessing, the detected activation maps looked

noisy and unrealistic. In the classical ICA approach, the corresponding activation maps were selected from the 64 independent component maps by using a correlation analysis between the time responses of the components and the stimuli. The detected activation at the second axial level of a representative subject, by using the SPM, the classical ICA with post-selection, and ICA-R are shown in the figures 7(a), (b) and (c), respectively. Activations detected by the classical ICA and ICA-R were similar to that of the SPM, but the detected activation by the ICA-R contained less spurious noise than others. As their MFLOPs compared in table 2, the ICA-R required much less computation resources and time than the classical ICA technique since only the task-related component map was detected without any pre- or post-processing, in a single process. Figure 8 shows the input stimulation used as the reference signal and the time response from a representative activated brain region detected using the ICA-R technique.

[Figure 7 is to be included here.]

[Figure 8 is to be included here.]

6 Discussion and Conclusion

The present ICA-R technique provides a general approach to extract an interesting subset of ICs in a mixture, in a single process if some information about the ICs is available, that can be formulated in the form of reference signals. The reference signals, traces of the sources, carry some information, such as peaks or zero crossings, of the desired sources, which were incorporated into the objective function by constraints representing such as non-Gaussianity of the sources or the closeness between the references and desired sources. The constrained optimization problem was then formulated in the cICA framework with some additional constraints: the uncorrelation to prevent different neurons from extracting to the same source and the unit variance of the estimated output components to avoid any instability of learning due to a dilation of the recovered signals.

As a Newton-like learning was adopted, the ICA-R algorithm was able to converge at least quadratically which is faster than the linear speed of the classical ICA. Further, the learning was simplified in the final formulation to speed the convergence by removing the matrix inversion. As the Hessian matrix is shown to be determinately positive definite, the learning stably converges to the minima of the objective function. Some critical parameters such as the nonlinear function for approximating negentropy, the closeness measure, and the threshold values may also affect the convergence; as shown, with the proper choice of the closeness measure and making the correct adjustments of the parameters, the algorithm can be made globally minimized to produce all the desired ICs.

The experiments demonstrated the advantages and superiority of the present algorithm compared to the earlier methods. The second-order approaches, using only the second-order statistics, failed to extract the sources close to the references, that are independent in the sense of higher-order statistics. As the higher-order statistical property - the negentropy - is used as the contrast function, the ICA-R technique accurately extracted the independent sources with high SNR and low PI. The classical ICA algorithm produces all ICs mixed in the inputs and therefore needs to select the desired signals by postprocessing, such as the correlation between the time responses and the references. Such a two-stage method estimates all the other uninteresting sources and thus is inefficient and lowers the quality of the separation of the desired components. The sources in the subspace extracted by the nonlinear PCA method are arbitrary and change from time to time and the extracted sources by the fastICA method are

always determined by their negentropy. In contrast, the present ICA-R algorithm projects the data onto a lower dimensional directed by the reference signals. Moreover, the extraction of desired sources by applying the ICA-R was performed in a single step learning process without any pre- and post-operations, such as pre-whitening or the explicit decorrelation, which are required by the other approaches.

The input stimulation involved in the experiments can be directly used as references for the analysis of fMRI data with the ICA-R which produces only the task-related components and discards all other non-task related components, interferences, and noise, simultaneously. This is very useful because only one or two components are task-related out of usually hundreds of components mixed in fMRI data. The classical ICA, on the other hand, produces all ICs which need to be selected for activated components. Evidently, the memory and computational requirements of the ICA-R are much less than those of the classical ICA. Furthermore, as shown in the synthetic fMRI experiment, the ICA-R always had the best performance on activation detection regardless of the choices of the *z*-value threshold. More importantly, the ICA-R technique enables the use of the temporal independence of the sources, which is practically prohibitive with the classical ICA. The detected activation in the fMRI experiments by the ICA-R algorithm contained most of the significant activation detected by the SPM technique. Also, the activation maps obtained with the ICA-R contained less spurious activation and noise. A comprehensive and qualitative comparison of the two techniques processing real fMRI data is impossible because the gold standards of the activation patterns for individual subjects are unavailable.

The ICA-R algorithm is only applicable for the analysis of fMRI data involving uncorrelated stimuli; orthogonalization of the correlated stimuli is required before using them as references. Also, there may be other components of the data, that are related to brain processes but may not correspond to the explicit stimuli used as references: for example, in the event-related experiments, memory retention phase is involved between the cue and probe phases. Nevertheless, the activation due to the hidden brain processes can be detected using the ICA-R if the corresponding reference signals can be identified and formed. Further, these reference stimuli need not to be exact, unlike in the hypothesis-driven models like SPM, as the rough templates of the corresponding stimulations are sufficient for the use of the ICA-R. Further exploration of ICA-R in real-world applications, such as fMRI, is worth pursuing.

Acknowledgement

The visual experiment data were collected while J. C. Rajapakse was a visiting scientist at the Max-Planck-Institute of Cognitive Neuroscience, Leipzig, Germany (1996-98). The authors wish to thank Mdm. Chan Soon Keng for proofreading the final manuscript.

References

[1] A. D. Back, A first application of independent component analysis to extracting structure from stock returns, Neural Systems 8 (4) (1997) 473–484.

[2] H.-M. Park, H.-Y. Jeong, T.-W. Lee, S.-Y. Lee, Subband-based blind signal separation for noisy speech recognition, Electronics Letters 35 (23) (1999) 2011–2012.

[3] J. Goldstein, J. Guerci, I. Reed, An optimal generalized theory of signal representation, in: Proceedings of 1999 IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 3, Phoenix, Arizona, USA, 1999, pp. 1357–1360.

[4] A. Bell, T. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neurocomputing 7 (1995) 1129–1159.

[5] S. Amari, A. Cichocki, H. Yang, A new learning algorithm for blind signal separation, in: Advances in Neural Information Processing Systems 8, MIT Press, Cambridge MA, 1996, pp. 752–763.

[6] T.-W. Lee, M. Girolami, T. Sejnowski, Independent component analysis using an extended informax algorithm for mixed sub-Gaussian and super-Gaussian sources, Neural Computation 11 (2) (1999) 409–433.

[7] J. Karhunen, J. Joutsensalo, Representation and separation of signals using nonlinear PCA type learning, Neural Networks 7 (1) (1994) 113–127.

[8] A. Hyvärinen, Fast and robust fixed-point algorithms for independent component analysis, IEEE Trans. on Neural Networks 10 (3) (1999) 626–634.

[9] M. Zibulevsky, B. A. Pearlmutter, Blind source separation by sparse decomposition, Neural Computation 13 (4).

[10] J. Luo, B. Hu, X. T. Ling, R. W. Liu, Principal independent component analysis, IEEE Trans. on Neural Networks 10 (4) (1999) 912–917.

[11] H.-M. Park, S.-H. Oh, S.-Y. Lee, A filter bank approach to independent component analysis and its application to adaptive noise cancelling, Neurocomputing 55 (3-4) (2003) 755–759.

[12] P. A. Bandettini, A. Jesmanowicz, E. C. Wong, J. S. Hyde, Processing strategies for time-course data sets in functional MRI of the human brain, Magnetic Resonance in Medicine 30 (1993) 161–173.

[13] K. J. Friston, Statistical parametric mapping, in: R. W. Thatcher, M. Hallett, T. Zeffiro, W. R. John, M. Huerta (Eds.), Functional Neuroimaging: Technical Foundations, Academic Press, 1994.

[14] W. Lu, J. C. Rajapakse, Constrained independent component analysis, in: Advances in Neural Information Processing Systems 13 (NIPS2000), MIT Press, 2000, pp. 570–576.

[15] J. C. Rajapakse, W. Lu, Unified approach to independent component networks, in: Second International ICSC Symposium on NEURAL COMPUTATION (NC'2000), Berlin, Germany, 2000.

[16] S. Amari, A. Cichocki, Adaptive blind signal processing - neural network approaches, Proceedings of the IEEE 86 (10) (1998) 2026–2048.

[17] T.-W. Lee, M. Girolami, A. J. Bell, T. J. Sejnowski, A unifying framework for independent component analysis, Computers and Mathematics with Applications 39 (11) (2000) 1–21.

[18] E. Oja, The nonlinear PCA learning rule in independent component analysis, Neurocomputing 17 (1989) 25–45.

[19] A. Cichocki, J. Karhunen, W. Kasprzak, R. Vigário, Neural networks for blind separation with unknown number of sources, Neurocomputing 24 (1-3) (1999) 55–93.

[20] P. Comon, Independent component analysis: A new concept?, Signal Processing 36 (1994) 287–314.

[21] A. Hyvärinen, New approximations of differential entropy for independent component analysis and projection pursuit, in: Advances in Neural Information Processing Systems 10 (NIPS*97), 1998, pp. 273–279.

[22] A. Hyvärinen, E. Oja, A fast fixed-point algorithm for independent component analysis, Neural Computation 9 (7) (1997) 1483–1492.

[23] M. Zibulevsky, Y. Zeevi, Extraction of single source from multichannel data using sparse decomposition, Tech. rep., Vision Research and Image Sciences Laboratory, Israel Institute of Technology (2001).

[24] X.-R. Cao, R.-W. Liu, General approach to blind source separation, IEEE Transactions on Signal Processing 44 (3) (1996) 562–571.

[25] A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, Inc., New York, 1991.

[26] M. Girolami, C. Fyfe, Extraction of independent signal sources using a deflationary exploratory projection pursuit network with lateral inhibition, Vision, Image and Signal Processing, IEE Proceedings 144 (5) (1997) 299–306.

[27] M. Girolami, C. Fyfe, Generalised independent component analysis through unsupervised learning with emergent bussgang properties, in: International Conference on Neural Networks, Vol. 3, 1997, pp. 1788–1791.

[28] D. P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, New York, 1982.

[29] P. P. Mitra, S. Ogawa, X. Hu, K. Ugurbil, The nature of spatiotemporal changes in cerebral hemodynamics as manifested in functional magnetic resonance imaging, Magnetic Resonance in Medicine 37 (1997) 511–518.

[30] M. McKeown, S. Makeig, G. Brown, T.-P. Jung, S. Kindermann, A. Bell, T. Sejnowski, Analysis of fMRI data by blind separation into independent spatial components, Human Brain Mapping 6 (1998) 160–188.

[31] J. C. Rajapakse, F. Kruggel, J. M. Maisog, D. Cramon, Modeling hemodynamic response for analysis of functional MRI time-series, Human Brain Mapping 6 (1998) 283–300.

[32] J. C. Rajapakse, J. Piyaratna, Bayesian approach to segmentation of statistical parametric maps, IEEE Transactions on Biomedical Engineering 48 (10) (2001) 1186–1194.

[33] M. McKeown, S. Makeig, G. Brown, T.-P. Jung, S. Kindermann, T. Sejnowski, Spatially independent activity patterns in functional magnetic resonance imaging data during the stroop color-naming task, Proceedings of the National Academy of Sciences 95 (1998) 803–810.

Algorithm	ICA-R		Second-Order		Classical ICA	
Output	y_1	y_2	y_1	y_2	y_1	y_2
SNR (dB)	22.86	21.95	4.50	3.07	17.76	16.49
PI	0.06		1.38		0.20	

Table 1: Comparison of the results of extracting two deterministic sources from the synthetic mixtures by using the algorithms of the ICA-R, second-order approach, and classical ICA; the signal-to-noise ratios (SNRs) of the two output components corresponds to the deterministic sources, c_2 and c_3; PI denotes the performance indices of the algorithms.

Algorithm		ICA-R	Classical ICA
MFLOPs	Synthetic Data	9.5	19.6
	Synthetic fMRI Data	16.1	12.4×10^3
	Real fMRI Visual Task Data	43.2	11.0×10^3

Table 2: Comparison of the computational complexity between the ICA-R and the classical ICA in terms of MFLOPs (Mega Floating points Operations) at three experiments of extracting desired signals from mixed synthetic one-dimensional data, synthetic fMRI data and real fMRI visual task data, respectively.

Figure Captions

Figure 1: Illustration of the experiment to extract two desired signals, deterministic sources c_2 and c_3: (a) five independent sources: Gaussian noise (c_1), periodic deterministic signals (c_2 and c_3), random signals (c_4 and c_5), (b) the mixture inputs, (c) the reference signals of periodic narrow pulses used for the extraction of the deterministic signals, (d) the outputs produced by the present ICA-R algorithm, (e) the outputs produced by the second-order approach, and (f) the selected two classical ICA output components closest to the two reference signals.

Figure 2: The comparison of PIs with iterations, when using classical ICA and ICA-R algorithms in extracting the desired signals from the mixtures of synthetic signals.

Figure 3: The two output components extracted in the experiments with the synthetic signals by applying (a) nonlinear PCA algorithm and (b) the fastICA algorithm.

Figure 4: The synthetic functional MR data having a SNR of -8.5 dB and a Gaussian spatial correlation of FWHM = 3.0 pixels. (a) Activated regions of four 9×9 square blocks, (b) the sixth image scan, and (c) the tenth image scan. Note that the gray contrasts of the figures (b) and (c) have been increased 80% from the actual image scans for easy visualization of the activated regions.

Figure 5: Comparison of performance of activation detection from the synthetic functional data at various *z*-value thresholds (1.5 ~ 3.5) by using the present ICA-R, the SPM and the classical ICA techniques: ROC curves.

Figure 6: Detected activation of the synthetic functional data by (a) using the ICA-R approach, (b) the SPM approach and (c) the classical ICA approach at z-value threshold of 2.75.

Figure 7: Task-related activation detected at the second axial slice of a representative subject performing the visual experiment, by (a) the SPM, (b) the classical ICA with post-selection, and (c) the ICA-R techniques. The significance values (z-values) of the activated voxels are shown color-coded.

Figure 8: The input stimulus used as the reference and the task-related time response from the activated brain voxels detected by the ICA-R algorithm at the second axial slice from a representative subject performing the visual task.