



Non-Gaussian Penalized PARAFAC Analysis for fMRI Data

Jingsai Liang¹, Jiancheng Zou² and Don Hong^{2,3*}

¹ Computer Science Department, Westminster College, Salt Lake City, UT, United States, ² College of Sciences, North China University of Technology, Beijing, China, ³ Computational Science Program, Middle Tennessee State University, Murfreesboro, TN, United States

OPEN ACCESS

Edited by:

Jianzhong Wang,
Sam Houston State University,
United States

Reviewed by:

Shuo Chen,
University of Maryland, United States
Yikai Wang,
Emory University, United States

*Correspondence:

Don Hong
don.hong@mtsu.edu

Specialty section:

This article was submitted to
Mathematics of Computation and
Data Science,
a section of the journal
Frontiers in Applied Mathematics and
Statistics

Received: 19 November 2018

Accepted: 18 July 2019

Published: 06 August 2019

Citation:

Liang J, Zou J and Hong D (2019)
Non-Gaussian Penalized PARAFAC
Analysis for fMRI Data.
Front. Appl. Math. Stat. 5:40.
doi: 10.3389/fams.2019.00040

Independent Component Analysis (ICA) method has been used widely and successfully in functional magnetic resonance imaging (fMRI) data analysis for both single and group subjects. As an extension of the ICA, tensorial probabilistic ICA (TPICA) is used to decompose fMRI group data into three modes: subject, temporal and spatial. However, due to the independent constraint of the spatial components, TPICA is not very efficient in the presence of overlapping of active regions of different spatial components. Parallel factor analysis (PARAFAC) is another method used to process three-mode data and can be solved by alternating least-squares. PARAFAC may converge into some degenerate solutions if the matrix of one mode is collinear. It is reasonable to find significant collinear relationships within subject mode of two similar subjects in group fMRI data. Thus, both TPICA and PARAFAC have unavoidable drawbacks. In this paper, we try to alleviate both overlapping and collinearity issues by integrating the characters of PARAFAC and TPICA together by imposing a non-Gaussian penalty term to each spatial component under the PARAFAC framework. This proposed algorithm can then regulate the spatial components, as the high non-Gaussianity can possibly avoid the degenerate solutions aroused by collinearity issue, and eliminate the independent constraint of the spatial components to bypass the overlapping issue. The algorithm outperforms TPICA and PARAFAC on the simulation data. The results of this algorithm on real fMRI data are also consistent with other algorithms.

Keywords: fMRI, ICA, TPICA, PARAFAC, non-Gaussian, penalty term, multi-subject data, tensor decompositions

1. INTRODUCTION

ICA is one of the most popular methods to analyze fMRI single or group data especially when the time courses are not available [1–4], such as the application of autism experiments [5]. Unlike general linear model (GLM) [6], ICA is a more general and totally data-driven method to decompose the mixed data into mixing matrix and source signals. One fundamental prerequisite requirement of ICA is that the distributions of the independent source signals must be non-Gaussian [7]. So under this assumption, because Gaussian variable has the largest entropy among all random variables with equal variance, one popular algorithm called fastICA [7] decomposes the data by projecting the data onto the unmixing directions, such that the projected data have the maximum non-Gaussianity. Such unmixing directions of the inverse of the mixing matrix are mutual orthogonal and can be calculated by the Newton algorithm. The projected data form independent source signals. In the application of fMRI data (temporal \times spatial), we usually call these source signals spatial components.

Many variants and improvements of ICA have developed in the past decade [8]. In the application of fMRI, the data usually have temporal-spatial information. We can use such information as the reference or constraint to improve the accuracy and robustness of the spatial components. The temporal information can be the time courses of design matrix in the experiment [9, 10]. The spatial information can be the atlas-defined masks [11, 12] or the cortex based information [13]. In some situation, the independent constraints are too strong to hold for some correlated spatial components. Subspace ICA [14–16] divides the spatial components into several groups and allows certain correlation of the components within the same groups. Multi-dimensional ICA [17] or topographic ICA [18] models the dependence through the spatial structure of the fMRI data, such as the neighboring voxels. Considering the sparsity character of the spatial components of the fMRI data, we can add the sparsity constraint to the fastICA to impose this character of the spatial components [19].

Besides the above developments of ICA to improve the accuracy and robustness of spatial components through the characters of themselves, it is very natural to consider the presence of the Gaussian noise in the fMRI data. The earlier work in Hyvärinen [20, 21] used a joint likelihood or the Gaussian moments to address this issue. Probabilistic ICA (PICA) [22] is another way to generalize the noise free ICA to incorporate the Gaussian noise. PICA assumes that the noise covariance is isotropic so that PICA could use a similar formula comparing with PPCA [23] to estimate and remove the noise before decomposing the data.

fMRI data usually contains multiple subjects as a group data. The heuristic way to process the group data is either combining the results of each subject or taking the average of all subjects. Beckmann and Smith [24] developed a three-mode ICA algorithm named tensorial probabilistic ICA (TPICA), which is suitable to decompose the group fMRI data into three modes: subject \times temporal \times spatial. Basically, as an extension of the PICA, TPICA not only decomposes fMRI data into the mixing matrix and spatial components by PICA, but also continues to decompose the mixing matrix to tensor product of time courses and subject loadings. In TPICA, the mode of spatial components has the priority to be processed, while the other two modes are estimated by one-rank approximations of mixed matrix. Based on TPICA, we can continue to refine the spatial components modeling by Gaussian mixture model and divide the tensor product of mixing matrix into different groups. This work has been completed in Guo [25] and Guo and Pagnoni [26]. The solution can be approximated by EM [27].

Parallel Factor Analysis (PARAFAC) is another popular method that processes three-mode data [28–30] and is becoming a new approach to process brain information and big data [31, 32]. Unlike TPICA, three modes are equally processed in PARAFAC. By the alternating least square (ALS) approach, one mode is estimated by least-squares from observed fMRI data while fixing other two modes. The three modes are iterated and alternated until the algorithm reaches convergence. PARAFAC does not require the independent constraint of the mode to achieve the solutions. One important property

of PARAFAC is that the solution is unique under proper conditions [28]. However because PARAFAC highly relies on the effective decomposition of least-squares, it may converge to some degenerate solutions if one mode of the data can not meet the full rank requirement of the least-squares. Models with perfect collinearity are not estimable because there is an infinite number of solutions that could fit the data equally well. While in fMRI data, it is reasonable to find significant collinear relationships of the subject loadings mode within two similar subjects. In addition, PARAFAC also may converge very slowly solved via ALS [24] due to inaccurate estimation of the number of the spatial components. Based on these reasons and some experiments, the authors of TPICA [24] concluded that TPICA is more robust and accurate than PARAFAC when it comes to estimating the underlying spatial components.

However, TPICA assumes that the spatial components of fMRI data are mutual independent. The contradiction is that it is very common to observe the overlapping of active regions of different spatial components in the spatial mode. In Stegeman [33] and Helwig and Hong [34], the authors noted that ICA is still the most effective way to deal with two-mode fMRI data. While in the three-mode group fMRI data, they argued that if fMRI data has trilinear structure, PARAFAC can achieve the unique decomposition under some proper conditions, such as having correctly determined number of spatial components. Without the consideration of the independent constraint among spatial components, PARAFAC can outperform TPICA in the presence of overlapping of activate regions in the spatial mode.

Both TPICA and PARAFAC have advantages and disadvantages in processing fMRI data. Although PARAFAC performs better in overlapping issues, it may suffer from collinearity issues, which are very common between similar subjects. Similarly, while TPICA is not affected by collinearity issue of the subject mode very much, it is not very effective to solve overlapping issues in three-mode fMRI data. Both of them cannot process three-mode fMRI data perfectly. However, if we can combine the advantages of PARAFAC and TPICA and eliminate their disadvantages, we may figure out a way to process fMRI group data to avoid both collinearity and overlapping issues. One typical method is to impose some constraints or penalty terms on three modes of PARAFAC [35, 36] to avoid the degenerate solutions. Alternatively we can combine PARAFAC and ICA by imposing the independent constraints in PARAFAC [37]. However, in the application of fMRI, one of the main tasks is to locate the active regions of the spatial components. We may only need to impose the constraints on each spatial component of the spatial mode [38]. In papers Martínez-Montes et al. [39, 40], the authors proposed a new penalty based on the information entropy for the spatial mode of the constrained PARAFAC analysis of resting EEG that allowed the identification in time, frequency and space of those brain networks with minimum spectral entropy.

In this paper, inspired by the method of penalty based PARAFAC, via imposing a non-Gaussian penalty term for the spatial components within the PARAFAC, we propose an algorithm that combines advantages and eliminates disadvantages of both TPICA and PARAFAC simultaneously. In

the case of collinearity issues, the degenerate solution of spatial components can be alleviated because the non-Gaussian penalty term can regulate each spatial component to be as non-Gaussian as possible. Meanwhile, this proposed algorithm can overcome the overlapping issue because it is still based on the PARAFAC, which does not need the independent constraint of the spatial components. Experiments of simulation data under different situations show that the proposed algorithm improves both accuracy and robustness compared with TPICA and PARAFAC. The results of this proposed algorithm on real fMRI data are also consistent with other methods.

This paper is organized as follows: In section 2, we will briefly review the related necessary algorithms. Then, we will propose our own non-Gaussian penalized algorithm in section 3. We design a series of experiments in section 4. The simulation and real data results will be presented in section 5. Finally, in section 6, we will discuss this proposed algorithm and potential future improvements.

2. MODEL REVIEW

In this section, we briefly describe the related models for processing group fMRI data. More detailed descriptions can be found in Guo and Pagnoni [26] and Bro [28], for example. Suppose that the observed fMRI group data Y consists of N subjects. Each subject contains V voxel samplings on T time points. Under the assumption that fMRI data has the trilinear structure, $Y(V \times T \times N)$ can be represented as a combination of three outer products:

$$Y = \sum_{k=1}^K a_k \otimes b_k \otimes c_k, \quad (1)$$

where K is the number of components and \otimes is the outer product. Vectors $a_k \in R^V$ is the spatial component, $b_k \in R^T$ is the time course, and $c_k \in R^N$ is the subject loading. For this three-way data, PARAFAC is one popular algorithm to decompose the data Y into three modes.

We can reshape the matrix Y to two dimensions via one mode [26, 28], such as:

$$Y_A = (C \odot B)A^t = MA^t, \quad (2)$$

where $A_{V \times K} = [a_1 a_2 \cdots a_K]$, $B_{T \times K} = [b_1 b_2 \cdots b_K]$, $C_{N \times K} = [c_1 c_2 \cdots c_K]$. $C \odot B = ((B \text{diag}(c_1))^t, \cdots, (B \text{diag}(c_K))^t)^t$ denotes Khatri-Rao product of C and B . Using this format, we can deem A as K independent spatial components of spatial mode and M as the mixing matrix. Then this equation becomes an ICA problem which is to decompose fMRI group data into mixing matrix M and spatial mode A .

It is not trivial to generalize ICA to group subjects analysis, naturally because different subjects in the group do not share the same independent components. Temporal concatenation of each subject data is the most popular way to organize the group data. Under this scheme, the group data can be easily processed as a single subject data and solved using same ICA method in each

iteration. Then, we can use some back-reconstruction methods [4] to rebuild each subject specific modes.

2.1. Tensorial PICA (TPICA)

We can use negentropy to characterize the non-Gaussianity of spatial components. For a random vector a , negentropy of a is defined as in Hyvarinen [7]:

$$J(a) = H(a_{\text{Gauss}}) - H(a), \quad (3)$$

where $H(a) = -\int f(a) \log f(a) da$ is the differential entropy of a , a_{Gauss} is Gaussian variable with zero mean and unit variance. One of the most important properties of the entropy is that a Gaussian variable a_{Gauss} has the largest entropy than any other variables with the same variance. Thus, negentropy of one random variable a is always negative and an indicator the non-Gaussianity of a . It is not easy to calculate the value of negentropy using formula (3). In general, we can estimate the negentropy approximately using this formula [7]:

$$J(a) \propto [E\{G(a)\} - E\{G(a_{\text{Gauss}})\}]^2, \quad (4)$$

where G is any nonquadratic function.

Without loss of generality, we assume M is a square matrix after pre-whitening step. FastICA [7] decomposes the data by projecting the data onto the rows of M^{-1} such that the projected data a_k have the maximum negentropy. M^{-1} is an orthogonal matrix and can be calculated by Newton algorithm. FastICA is noise free and needs squared mixing matrix. Probabilistic model [22] PICA is one way to incorporate the Gaussian noise:

$$Y_A = MA^T + E, \quad (5)$$

where E denotes concatenated isotropic Gaussian noise matrix and time series data at each voxel follows $N(0, \sigma^2 I)$.

Let $\tilde{Y} = U_K(N\Lambda_K)^{\frac{1}{2}} V_K$ be its rank- K singular value decomposition (SVD). By using the property of equal variance at both sides, we can obtain the explicit solution of Equation (5) [22].

$$\tilde{M} = U_K(\Lambda_K - \sigma^2 I_K)^{\frac{1}{2}} Q^T, \quad (6)$$

$$\tilde{\sigma} = \frac{1}{NT - K} \sum_{i=K+1}^{NT} \lambda_i, \quad (7)$$

where λ_i denotes the diagonals of Λ , and Q denotes the rotation matrix coming from an ICA algorithm.

The TPICA algorithm continues to model M by the Khatri-Rao product of subject mode C and temporal mode B . At first, TPICA uses PICA to estimate the spatial mode A and mixing matrix M . Then, TPICA rebuilds the temporal components B and subject loadings C from mixing matrix M . If we reshape the i th column M as an $N \times T$ matrix m_i and calculate its SVD:

$$m_i = U_m \Lambda_m V_m^T, \quad (8)$$

then we can approximate $c_i \odot b_i \approx m_i$ by the rank-1 approximation of m_i . TPICA iterates these two steps by initializing the input of PICA first and then using the approximation results for the next step until convergence.

2.2. PARAFAC

Harshman [29] proposed the PARAFAC method for decomposing the three-mode data. The goal of PARAFAC is to minimize the following error:

$$A, B, C = \operatorname{argmin}_{A, B, C} \| Y - \sum_{k=1}^K a_k \otimes b_k \otimes c_k \|^2. \quad (9)$$

PARAFAC treats all three modes equally and does not incorporate any spatial or temporal information. This means that PARAFAC itself does not restrain the independence of the spatial components, which is one key constraint in TPICA.

The most attractive feature of PARAFAC is the uniqueness of its solution under the proper conditions [28]. Regardless of the scaling, the decomposition of A, B, C is unique if

$$k_A + k_B + k_C \geq 2K + 2, \quad (10)$$

where k is the Kruskal rank or k -rank of the matrix, which is the largest number k such that every subset of k columns of this matrix is independent.

Alternating Least Squares (ALS) is the basic method to solve this problem. Actually, ALS iterates least-square estimation for one of A, B, C while fixing the other two matrices. The iterative pseudo code is as following:

$$\begin{aligned} A &= Y_A Z (Z^T Z)^{-1} \text{ where } Z = C \odot B, \\ B &= Y_B Z (Z^T Z)^{-1} \text{ where } Z = C \odot A, \\ C &= Y_C Z (Z^T Z)^{-1} \text{ where } Z = B \odot A, \end{aligned} \quad (11)$$

where Y_\bullet is the reshape of Y according to the modes.

ALS is a very attractive algorithm for solving PARAFAC because it ensures an improvement of the solution in each iteration. It will converge to the global minimum in most well-behaved problem [28]. A common stopping criterion is to stop iterations when the relative change between two iterations is less than a predefined small value.

PARAFAC is sensitive to the estimated number of source signals and the rank of the matrices of three modes. Some constraints can be added to PARAFAC model to avoid degenerate solutions. In some applications, imposing the related meaningful constraint can improve the accuracy and interpretation of the solutions. Without loss of generality [38], if we want to add one penalization $P(A)$ on the mode A , we can modify the first equation from formula (11) as:

$$\tilde{A} = \operatorname{argmin}(\| Y_A - Z A^T \|^2 + \lambda P(A)). \quad (12)$$

In the formula (12), the penalty term is restricted on the whole matrix A . However, in the application of fMRI data, we only want to regulate each spatial component which is the column a of A . We can modify formula (12) to impose penalty on each a_i , column of A .

$$\tilde{A} = \operatorname{argmin}(\| Y_A - Z A^T \|^2 + \lambda \sum_{i=1}^K P(a_i)). \quad (13)$$

Here the positive penalty parameter λ balances the weights of two terms. More importantly, this penalized PARAFAC will still hold the property of convergence if we could make sure the solution of the object function (13) can be improved in each iteration as well. With the penalty, PARAFAC no longer treats the modes equally any more. It can incorporate the related information into one specific mode to improve the quality of the corresponding components.

3. PROPOSED NON-GAUSSIAN PENALIZED PARAFAC

In the ICA analysis of fMRI data, the fundamental assumption is that the components of spatial matrix A follow non-Gaussian distribution. The core of the fastICA algorithm is to find a direction to project fMRI data Y so that the projected spatial component a is as non-Gaussian as possible. In a similar way, inspired by formula (13), we can impose the non-Gaussian penalty on each column a of spatial components A to increase their non-Gaussianities. On the other hand, non-Gaussianity can be approximated by negentropy. Thus, we can use the reciprocal of negentropy of a_j to form the penalty term $P(a_j)$. Therefore, in order to achieve minimal value of object function (13), a_j needs to be considered as non-Gaussian signal while approaching the least square estimates. As for the collinearity issue in the subject component, the possible degenerate solution of ALS could be avoided by using this penalty term as well.

We can decompose the product of two matrices A and Z to the summation of k products of their corresponding columns as follows.

$$\begin{aligned} & \| Y_A - Z A^T \| + \lambda \sum_{i=1}^K P(a_i) \\ &= \| Y_A - \sum_{i=1, i \neq j}^K z_i a_i^T - z_j a_j^T \| + \lambda \sum_{i=1, i \neq j}^K P(a_i) + \lambda P(a_j) \\ &= \| Y_j - z_j a_j^T \| + P_j + \lambda P(a_j), \end{aligned} \quad (14)$$

where z_j is j th column of Z , $Y_j = Y_A - \sum_{i=1, i \neq j}^K z_i a_i^T$, and $P_j = \lambda \sum_{i=1, i \neq j}^K P(a_i)$. The solution of column a_j by least-squares is $\hat{a}_j = \frac{Y_j^T z_j}{z_j^T z_j}$. We can iterate this procedure to estimate matrix A until it converges. We can have the following lemma with a similar proof shown in the paper [38].

Lemma 1. Minimization task of function related to a_j :

$$\min \| Y_j - z_j a_j^T \| + P_j + \lambda P(a_j) \quad (15)$$

is equivalent to minimization task of function related to a_j :

$$\min \| a_j - \hat{a}_j \| + \bar{\lambda} P(a_j), \quad (16)$$

where \hat{a}_j is least-squares solution $\frac{Y_j^T z_j}{z_j^T z_j}$ and $\bar{\lambda} = \frac{\lambda}{z_j^T z_j}$.

Proof. Let $\hat{a}_j = \frac{Y_j^T z_j}{z_j^T z_j}$. Then

$$\begin{aligned} & \operatorname{argmin}_{a_j} \| Y_j - z_j a_j^T \| + P_j + \lambda P(a_j) \\ &= \operatorname{argmin}_{a_j} \| (Y_j - z_j \hat{a}_j) + z_j \hat{a}_j - z_j a_j^T \| + P_j + \lambda P(a_j) \\ &= \operatorname{argmin}_{a_j} \text{constant} + z_j^T z_j \| a_j - \hat{a}_j \| + \lambda P(a_j) \\ &= \operatorname{argmin}_{a_j} \| a_j - \hat{a}_j \| + \bar{\lambda} P(a_j), \end{aligned} \tag{17}$$

where $\bar{\lambda} = \frac{\lambda}{z_j^T z_j}$.

To minimize object function (16) is equivalent to increase the negentropy of each spatial component a_j and keep the regulated component a_j not too far away from their initial value \hat{a}_j . However, we must standardize vector a_j before we substitute a_j into the formula of negentropy [7].

Lemma 2. Minimization task of function related to a_j

$$\min \| a_j - \hat{a}_j \| \tag{18}$$

is equivalent to minimization task of function related to a_j :

$$\min \| \bar{a}_j - \bar{\hat{a}}_j \|, \tag{19}$$

where $\bar{a}_j = \frac{a_j - \text{mean}(a_j)}{\text{std}(a_j)}$, $\bar{\hat{a}}_j = \frac{\hat{a}_j - \text{mean}(\hat{a}_j)}{\text{std}(\hat{a}_j)}$.

Proof

$$\begin{aligned} & \operatorname{argmin}_{a_j} \| \bar{a}_j - \bar{\hat{a}}_j \| \\ &= \operatorname{argmin}_{a_j} \| \frac{a_j - \text{mean}(a_j)}{\text{std}(a_j)} - \frac{\hat{a}_j - \text{mean}(\hat{a}_j)}{\text{std}(\hat{a}_j)} \| \\ &= \operatorname{argmin}_{a_j} \| \bar{a}_j - \bar{\hat{a}}_j \| * \frac{1}{\text{std}(\hat{a}_j)} \\ &= \operatorname{argmin}_{a_j} \| a_j - \hat{a}_j \|. \end{aligned} \tag{20}$$

After combing all these equivalent functions, we can convert object function (12) into the following non-Gaussian penalized formula:

$$\min \| \bar{a}_j - \bar{\hat{a}}_j \|^2 + \bar{\lambda} \frac{1}{(E(G(\bar{a}_j)) - E(G(v)))^2}, \tag{21}$$

where v is Gaussian signal with zero mean and unit variance, G is a nonquadratic function, $G(u) = \frac{1}{p} \log \cosh(pu)$ and its derivative $g(u) = \tanh(pu)$, where $p = 1$.

It is difficult to find a closed-form solution of function (21). Here we use backtracking gradient descent algorithm to find the local minimum of function (21). The derivative of the object function (21) for \bar{a}_j is the deepest increasing direction d .

$$d = 2(\bar{a}_j - \bar{\hat{a}}_j) - \frac{2\bar{\lambda}g(\bar{a}_j)}{V(E(G(\bar{a}_j)) - E(G(v)))^3}. \tag{22}$$

Choose the \hat{a}_j as the initial value of a_j and move the \bar{a}_j slightly along the negative direction d by step size α to decrease the value of object function (21). We may need to shrink the step size α by γ if necessary using $\gamma \in (0, 1)$. Afterward, we need to re-standardize the new value of \bar{a}_j and move it again along with the current new direction d . Iterate this procedure until the difference between previous d and current d are very close. Here is the pseudocode of this backtracking line search algorithm:

Choose $\alpha > 0$, $\gamma \in (0, 1)$.
Repeat until the value of object function (21) is reduced by new \bar{a}_j ,
where $\bar{a}_j = \bar{\hat{a}}_j + \alpha * \frac{d}{\text{norm}(d)}$, $\alpha = \gamma * \alpha$.

This proposed algorithm can overcome the overlapping issue because it is still based on the PARAFAC framework, which does not need the independent constraint of the spatial components. However, the difference is that under this proposed algorithm framework, three modes are no longer equally processed anymore. Spatial mode will be calculated column by column and each column will be regulated by non-Gaussian penalty.

Now, we are ready to outline the proposed algorithm.

Initialization:

1. Reshape three-dimensional Y to two-dimensional Y_a, Y_b, Y_c .
2. Initialize $A, B, C, \eta_1, \eta_2, \eta_3, \epsilon_d, \epsilon_A, \epsilon_Y$.

Iterative Procedure:

1. $B = Y_b Z(Z^T Z)^{-1}$ where $Z = C \odot A$
2. $C = Y_c Z(Z^T Z)^{-1}$ where $Z = B \odot A$
3. First Sub Iterative Procedure. Set column number $j = 1$.
(Calculate columns a_j of A one by one):
 - 3.1 $\hat{a}_j = Y_j z_j(z_j^T z_j)^{-1}$ where z_j is j th column of $Z = C \odot B$
 - 3.2 Second Sub Iterative Procedure.
(Non-Gaussian Penalty of a_j):
 - 3.2.1 Calculate a_j by backtracking line search.
 - 3.2.2 Calculate $\eta_1 = \frac{\text{norm}(\text{previous } d) - \text{norm}(d)}{\text{norm}(d)}$. If $\eta_1 > \epsilon_d$, return to step 3.2.1. Otherwise continue to step 3.3.
 - 3.3 If $j = K$, continue to step 3.4. Otherwise set $j = j + 1$ and return to step 3.1.
 - 3.4 Calculate $\eta_2 = \frac{\text{norm}(\text{previous } A) - \text{norm}(A)}{\text{norm}(A)}$. If $\eta_2 > \epsilon_A$, set $j = 1$ and return to step 3.1. Otherwise continue to step 4.
4. $\text{err} = Y_a - Z A^T$, Calculate $\eta_3 = \frac{\text{norm}(\text{previous err}) - \text{norm}(\text{err})}{\text{norm}(\text{err})}$. If $\eta_3 > \epsilon_Y$, return to step 1. Otherwise stop this algorithm.

3.1. Convergence Analysis

This proposed algorithm is composed of three nesting loops. The first loop is PARAFAC framework. The second loop is to obtain least-squares solution of each component a_j of matrix A . The third loop is to optimize the current component a_j using the non-Gaussian penalty. Therefore, this algorithm calculates three relative errors η_1, η_2, η_3 inside each loop to compare with predefined threshold parameters $\epsilon_d, \epsilon_A, \epsilon_Y$. If all relative errors $\eta_i, i = 1, 2, 3$ are less than predefined parameter $\epsilon_i, i = 1, 2, 3$, this algorithm converges to a stable decomposition of fMRI data Y .

ALS will converge in most well-behaved problem if we can ensure that the solution can be improved in each iteration [28]. In the third loop, we use backtracking line search to ensure that the new column a_j can reduce the value of the object function (21). In the first and second iterations, we all apply the alternating method to optimize each component matrix or each column of one matrix A via ordinary linear square algorithm. Thus, both in theoretical analysis and following practical experience, our proposed algorithm converges to a stable decomposition of Y .

3.2. Parameters Selection

Multiple parameters are needed to be defined properly including penalty parameter λ , step size parameter α , γ , three threshold parameters $\epsilon_d, \epsilon_A, \epsilon_Y$, etc. Appropriate values of these parameters are needed to lead to a successful decomposition. We use the grid search method to choose the best combination of values of these parameters.

We set ϵ_Y as $1e - 6$, which is same as the default value of ϵ used in the convergence criterion of PARAFAC from the N -way toolbox, available at www.models.life.ku.dk/source/nwaytoolbox/. We define 0.9 as the value of γ because we do not want to shrink the step size parameter too much in each loop. The most important parameter among them is the penalty parameter λ because λ is the key to

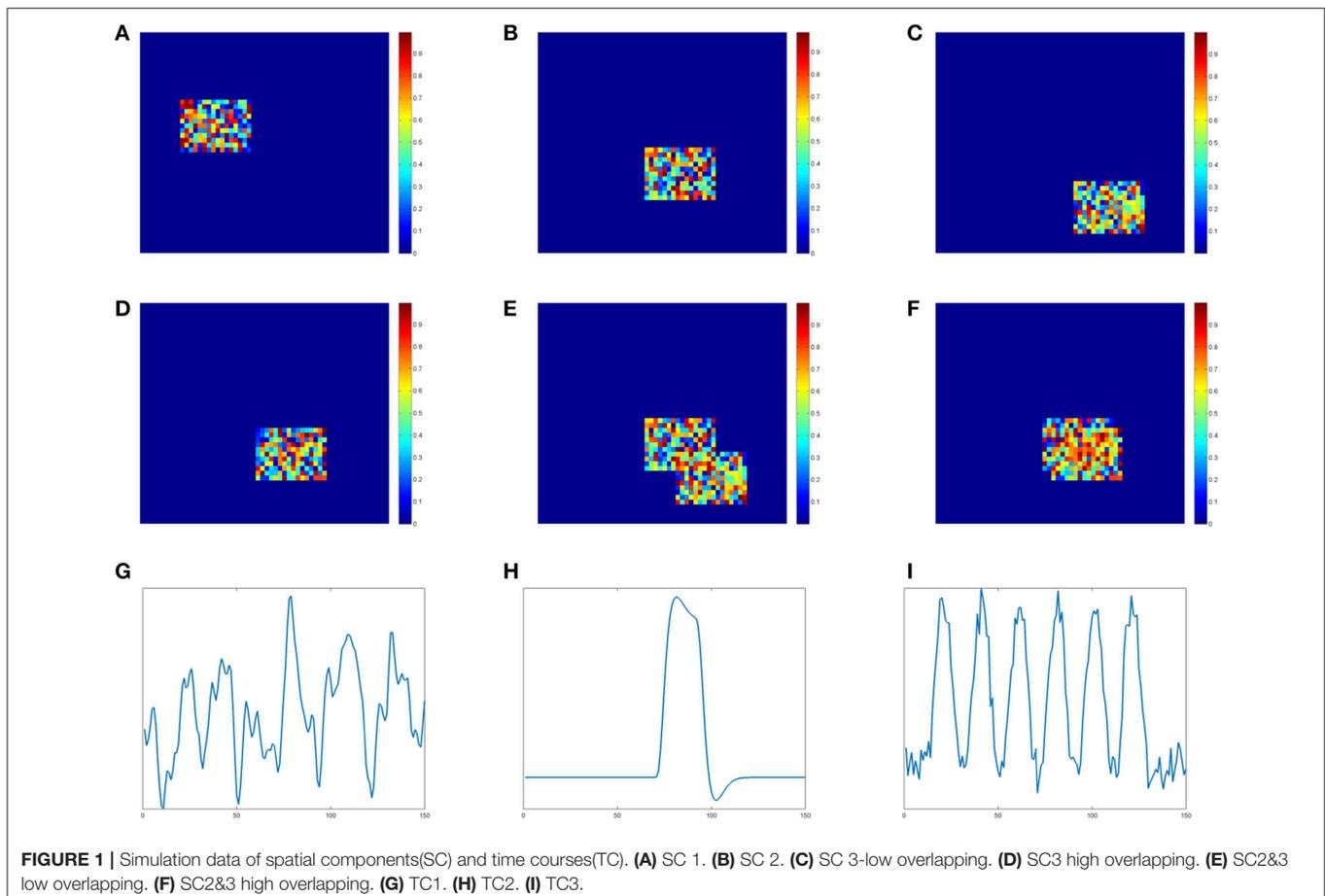
balance the non-Gaussian penalized weight and the similarity between initial component \hat{a}_j and improved component a_j . Here are our initial choices for these parameters: $\lambda = [0.5, 1, 5, 10]$; $\alpha = [0.1, 0.2]$; $\epsilon_d = [1e - 2, 1e - 3, 1e - 4]$; $\epsilon_A = [1, 0.9]$. Then, we run the algorithm for each combination of these values on a simulated data in next section. By comparing the performances of these combinations in a such grid search way, we define λ as 1, α as 0.1, ϵ_d as $1e - 3$, and ϵ_A as 1.

4. EXPERIMENTAL DESIGN

We verify the algorithm on both simulation data and real fMRI group data. For simulation data, we compare the results of four algorithms (TPICA, PARAFAC, Back Construction Algorithm [41], and proposed Non-Gaussian Penalized PARAFAC) with the ground truth design. For the real fMRI group data, we compare the result of our algorithm with the result in paper [41].

4.1. Simulation Data

We generate four two-dimensional matrices of size 46×56 to represent slices of brain voxels. Each matrix simulates one spatial component (SC) and contains an active region with size 11×16 . In **Figures 1A–D** shows these four simulated spatial components and their active regions. The voxel value of the background is 0.



The voxel value at the active region is randomly sampled from uniform distribution [0,1]. In order to show the different effects of low and high overlapping SCs, we design two copies of the third SC. The one in **Figure 1C** is the low overlapping case of SC3 and SC2, while the one in **Figure 1D** is the high overlapping case of SC3 and SC2. Two overlapping results are demonstrated visually in **Figures 1D,E**. If we define the ratio between the shared region and the active region as the overlapping rate, the rate of the low one in **Figure 1E** is 20% and the rate of the high one in **Figure 1F** is 70%. SC1, SC2 and one SC3 together form the matrix *A* with three spatial components.

We simulate time courses (TC) using the convolution of the stimulus functions with the hemodynamic response function (HRF) which is generated by SPM function¹ `spm_hrf()`. We use block design pattern and a single peaked function as the stimulus functions for TC2 and TC3, respectively. TC1 is sampled from time course of real fMRI data [41]. Each TC contains 150 time points and is shown in **Figures 1G–I**, respectively. These three TCs together form the matrix *B* with three times courses.

In order to verify the advantage of our proposed algorithm in processing collinearity issue of the subject loading in group ICA, we provide two subject loading matrices *C* in Equation (23). *C*₁ is a full rank squared matrix with a low condition number which represents a low collinearity case, while the condition number of *C*₂ is much higher, which represents a high collinearity case. Each matrix represents 10 different subjects.

$$C_1 = \begin{pmatrix} 2 & 2 & 3 \\ 3 & 3 & 3 \\ 1 & 1 & 1 \\ 1 & 3 & 3 \\ 1 & 1 & 3 \\ 1 & 3 & 3 \\ 1 & 2 & 1 \\ 2 & 2 & 1 \\ 2 & 1 & 1 \\ 2 & 1 & 3 \end{pmatrix}, C_2 = \begin{pmatrix} 2 & 2 & 3 \\ 3 & 3 & 3 \\ 1 & 1 & 1 \\ 3 & 3 & 3 \\ 1 & 1 & 3 \\ 3 & 3 & 3 \\ 2 & 2 & 1 \\ 2 & 2 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 3 \end{pmatrix}. \quad (23)$$

We also consider the noise effect to evaluate the robustness of the algorithm. We define the SNR of the data *Y* as

$$SNR = \frac{\|(C \odot B)A^T\|_F}{\|E\|_F}, \quad (24)$$

where *E* is a matrix with same shape of *Y*. Each element of *E* is a random Gaussian noise. We update noised *Y* using $Y = Y + E$. The mean of *E* is 0. The standard deviation σ of *E* has two values: 0.2 and 0.5. Based on the formula (24), the corresponding values of SNR are about 1.5 and 0.6. Note that a low SNR value means high noise level.

In **Table 1**, we design eight different tests against three indicators: noise level, overlapping level and collinearity. Each indicator has two contrasting values: low and high. Noise level is evaluated by SNR of the data *Y*, overlapping level is evaluated by overlapping rate of the shared region, and

TABLE 1 | Experimental List of Simulation.

	Noise level (SNR)	Overlapping level (overlapping rate)	Collinearity level (condition number of <i>C</i>)
Experiment A	Low (1.5)	High (70%)	High (117)
Experiment B	Low (1.5)	High (70%)	Low (11)
Experiment C	Low (1.5)	Low (20%)	High (117)
Experiment D	Low (1.5)	Low (20%)	Low (11)
Experiment E	High (0.6)	High (70%)	High (117)
Experiment F	High (0.6)	High (70%)	Low (11)
Experiment G	High (0.6)	Low (20%)	High (117)
Experiment H	High (0.6)	Low (20%)	Low (11)

It contains three indicators: noise level, overlapping level and collinearity level. Each of them has both low value and high value.

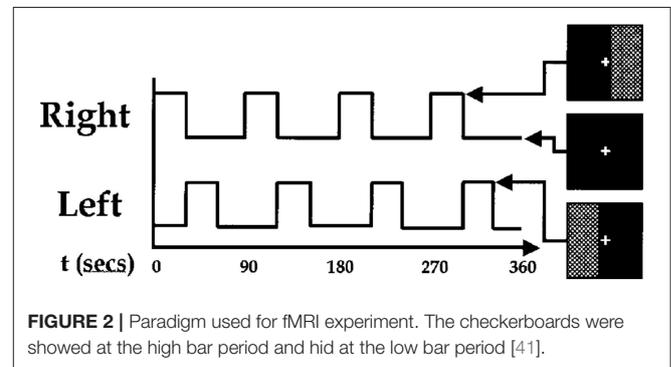


FIGURE 2 | Paradigm used for fMRI experiment. The checkerboards were shown at the high bar period and hid at the low bar period [41].

level of collinearity is evaluated by the condition number of subject loading matrix *C*. In the next section, we will show the comparison method of these three algorithms and the experimental results.

4.2. Real fMRI Group Data

We use the data from paper [41] completed at Johns Hopkins University. Data from nine normal subjects were acquired on a Philips 1.5T Scanner. Functional scans were acquired with an echo planar sequence (64 × 64, flip angle = 90, TR = 1 s, TE = 39 ms) over a 6-min period for a total of 360 time points. A visual paradigm was designed in which an 8 Hz reversing black and white checkerboard was presented intermittently in the left and right visual fields. The checkerboards were shown for 30 s and were resumed after 60 s. Subjects were focusing on a central cross on the checkerboard during the entire 6 min. The paradigm is depicted in **Figure 2**. We only have partial data of three subjects instead of nine subjects in this experiment.

5. RESULTS

5.1. Simulation Data

We compare four algorithms including TPICA, PARAFAC, Back Construction Algorithm [41], and the proposed Non-Gaussian Penalized PARAFAC. In experiments A–H, we run each algorithm ten times and compare results of SC with the ground

¹SPM. <https://www.fil.ion.ucl.ac.uk/spm/>

truth SC in **Figure 1** using congruence coefficient ρ [42], where

$$\rho = \frac{a^T b}{\sqrt{(a^T a)(b^T b)}}, \text{ given column vectors } a, b. \quad (25)$$

Here the value ρ can be used to measure the similarity between the restored component and the truth component. **Table 2** shows the means and variances of ρ of spatial sources and **Table 3** shows the means and variances of ρ of time courses of each algorithm in each experiment. **Figure 3** draws the content of **Table 2** in the format of error bars. Then we can compare these four algorithms intuitively.

In **Figure 3**, the results of four algorithms in each experiment are showed in one separate subfigure. The horizontal axis shows four algorithms in the order of TPICA, PARAFAC, Back Construction Algorithm, and the proposed algorithm from left to right with the color of green, blue, red, and magenta, respectively. The vertical axis represents the value of ρ . Four vertical error bars represent the intervals of one standard deviation away from the means of ρ .

From **Figure 3**, it is clear to see that all the assumptions in the previous sections are verified in these eight experiments. (1) The proposed algorithm is the best one compared with TPICA and PARAFAC in all eight experiments. Not only is the mean value of proposed algorithm the highest, but the standard deviation of the proposed algorithm is also the smallest. A higher mean value means a better decomposition, and a lower standard deviation indicates a more stabilized algorithm. Thus, this proposed algorithm overcomes both the collinearity and overlapping issues in this simulation. (2) High noise level can cut down the performances of all algorithms. Experiments (a)–(d) with high level of noise are generally better than experiments (e)–(h) with low level of noise. If the noise level is set to be even higher, it would be very difficult for any algorithm to decompose the data into meaningful components. (3) High collinearity level can dramatically reduce the performance of PARAFAC. Experiments (c) and (g) are typical cases with high collinearity level and low overlapping level. We can see that, in these two experiments, PARAFAC permorms the worst and is not very stable. Adding a non-Gaussian penalty term on the spatial mode can conquer this intrinsic drawback of PARAFAC. (4) High overlapping level can decrease the performance of TPICA although it is still a quite robust algorithm. Experiments (b) and (f) are testing on high overlapping and low collinearity level. TPICA works clearly the worst this time, especially in experiment (d). The independent requirements of ICA hinder any modifications based on itself to deal with the overlapping issue. Thus, we need to modify the algorithm based on PARAFAC in order to avoid this issue.

Figures 4–6 show the decomposition results of the first run of experiments A, B, and C. These figures again verify the above conclusions visually. The time courses in **Figures 4–6** are standardized by subtracting their means and divided by their standard deviations. **Figure 4** shows the results in the case of low noise, high overlapping and high collinearity. We see that the active regions of SC2 and SC3 of TPICA can not detach

TABLE 2 | Experimental Results of the mean and std of ρ of spatial sources in all four algorithms.

	TPICA	PARAFAC	Back construction	Non-Gaussian penalized PARAFAC
Mean(Std) of Congruence Coefficient ρ of Spatial Sources				
A	0.9643 (0.0101)	0.8920 (0.0648)	0.8882 (0.0103)	0.9837 (0.0033)
B	0.9630 (0.0103)	0.9981 (0.0001)	0.8870 (0.0092)	0.9982 (0.0001)
C	0.9970 (0.0005)	0.9369 (0.0499)	0.9523 (0.0587)	0.9905 (0.0087)
D	0.9969 (0.0004)	0.9981 (0.0001)	0.9503 (0.0592)	0.9982 (0.0001)
E	0.9581 (0.0079)	0.9264 (0.0669)	0.8620 (0.0512)	0.9756 (0.0060)
F	0.9570 (0.0088)	0.9884 (0.0006)	0.8708 (0.0204)	0.9897 (0.0005)
G	0.9886 (0.0006)	0.9084 (0.0797)	0.9121 (0.0830)	0.9721 (0.0220)
H	0.9860 (0.0009)	0.9880 (0.0006)	0.9523 (0.0055)	0.9895 (0.0005)

This table shows the proposed Non-Gaussian Penalized PARAFAC performs best in all eight experiments and is the most robust one in general.

TABLE 3 | Experimental Results of the mean and std of ρ of time courses in all four algorithms.

	TPICA	PARAFAC	Back construction	Non-Gaussian penalized PARAFAC
Mean (Std) of Congruence Coefficient ρ of Time Courses				
A	0.9619 (0.0230)	0.8982 (0.0685)	0.8198 (0.0068)	0.9923 (0.0013)
B	0.9540 (0.0256)	0.9999 (0.0000)	0.8169 (0.0079)	0.9999 (0.0000)
C	0.9971 (0.0010)	0.9460 (0.0537)	0.8346 (0.0721)	0.9893 (0.0247)
D	0.9973 (0.0009)	0.9999 (0.0000)	0.8252 (0.0829)	0.9999 (0.0000)
E	0.9646 (0.0240)	0.9387 (0.0662)	0.7996 (0.0711)	0.9837 (0.0242)
F	0.9693 (0.0210)	0.9994 (0.0001)	0.8126 (0.0082)	0.9994 (0.0001)
G	0.9968 (0.0010)	0.9322 (0.0568)	0.7964 (0.1098)	0.9626 (0.0543)
H	0.9962 (0.0012)	0.9994 (0.0001)	0.8565 (0.0060)	0.9995 (0.0000)

This table shows the proposed Non-Gaussian Penalized PARAFAC performs best in all eight experiments and is the most robust one in general.

from each other very well. One active region of SC includes a light shaded region from another SC. **Figure 5** shows the results in the case of low noise, high overlapping and low collinearity. We can see that TPICA performs worst under this situation. The results of time course in TPICA are not very consistent with the truth values and the spatial components in TPICA still have obvious shaded regions around the truth components. **Figure 6** shows the results in the case of low noise, low overlapping and high collinearity. This time both TPICA and the proposed algorithm outperform PARAFAC. PARAFAC leads to the degenerate solutions. The first spatial component in the results of PARAFAC is totally meaningless.

5.2. Real fMRI Group Data

We embed our algorithm into the GIFT package². The final results of the algorithm are summarized in **Figure 7**, which

²GIFT. <http://mialab.mrn.org/software/gift/index.html>

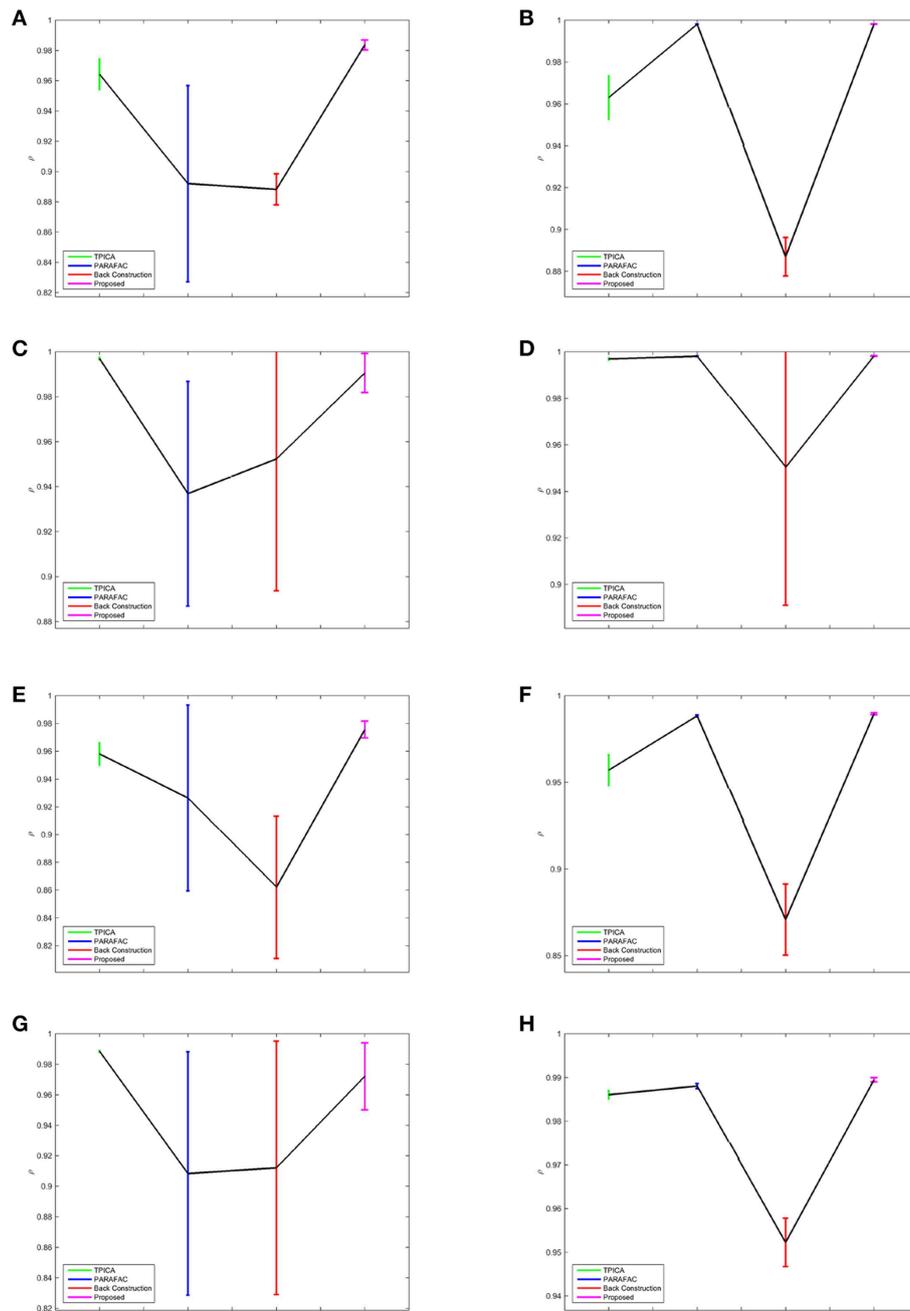
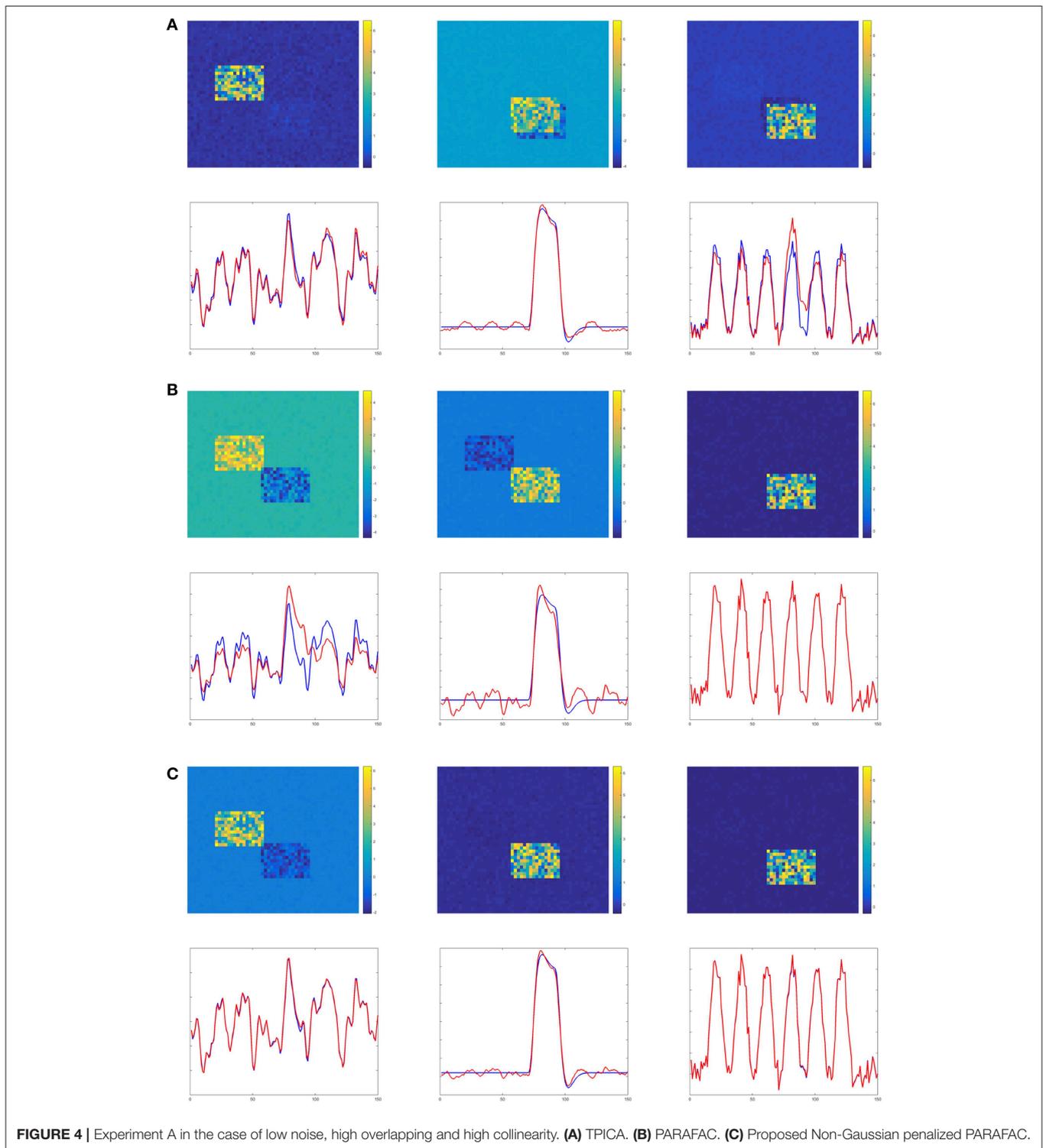


FIGURE 3 | Error Bars of the Experimental Results. The colors of TPICA, PARAFAC, Back Construction Algorithm, proposed Non-Gaussian Penalized PARAFAC from left to right are green, blue, red, and magenta. The vertical axis represents the mean of ρ . The vertical colorful bar represents one std away from the mean value. **(A)** Experiment A. **(B)** Experiment B. **(C)** Experiment C. **(D)** Experiment D. **(E)** Experiment E. **(F)** Experiment F. **(G)** Experiment G. **(H)** Experiment H.

is generated by GIFT. The threshold value is 2.0 and the slice range is -52:4:8. The activated spatial map shows right visual cortex (blue), left visual cortex (red); a transiently task-related component (TTR, green) in bilateral occipital/parietal cortex. These results are consistent with the results in Calhoun et al. [41], while here we only use three subjects instead of nine.

The head part of **Figure 7** shows three time courses of the below components. The blue one is the time course of right visual cortex and the red one is the time course of left visual cortex. The peaked and flat periods of two time courses are equivalent to the paradigm in **Figure 2** used to control the showing and disappear of the checkerboards in the left and right visual fields.



6. DISCUSSION

In this paper, we successfully alleviate both overlapping and collinearity issues aroused by ICA and PARAFAC by adding a non-Gaussian penalty term to the spatial mode calculation

of the PARAFAC. This algorithm can also be deemed as the combination of the characters of ICA and PARAFAC.

The strategy of the fastICA is to project fMRI data onto several orthogonal directions to maximize the non-Gaussianity of the projected data, which are the spatial components.

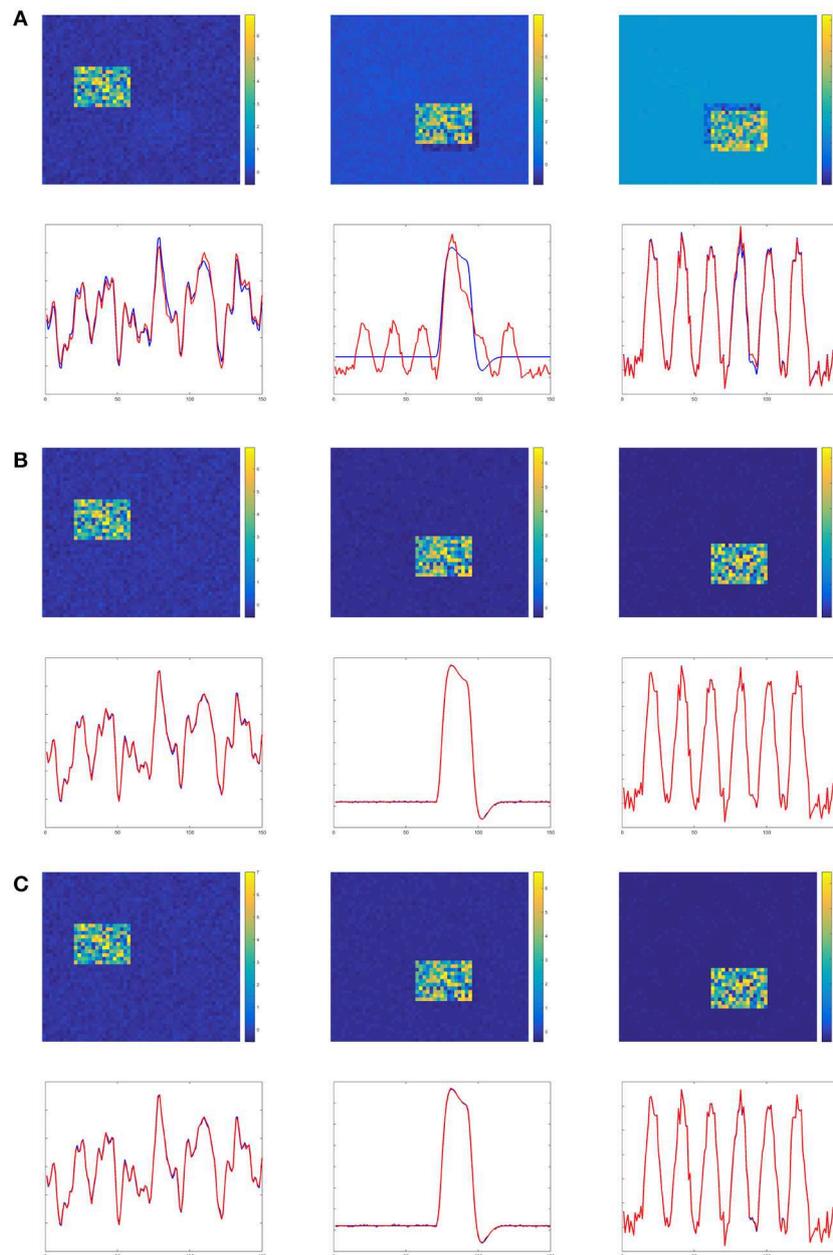


FIGURE 5 | Experiment B in the case of low noise, high overlapping and low collinearity. It shows TPICA suffers from the overlapping issue. **(A)** TPICA. **(B)** PARAFAC. **(C)** Proposed Non-Gaussian penalized PARAFAC.

So the assumption of non-Gaussianity is the key character of the spatial components to identify and extract them. The potential degenerate solutions of PARAFAC can be regulated toward the directions (22) of high non-Gaussianity to maximize the proposed object function (21). By this non-Gaussian penalty term, the degenerate solutions could be improved to a good solutions. On the other hand, in the overlapping situation, mutual orthogonal directions calculated by ICA cannot distinguish the correlated spatial components very well. This proposed algorithm is still based

on the PARAFAC framework, which does not need the independent constraint of the spatial components. Thus, it can separate the correlated components better than the ICA method.

This proposed algorithm uses the result of fastICA as the initial input. In this way, the algorithm can converge faster and run more robustly than random initial values. Additionally, this better result obtained by this proposed algorithm cannot be achieved by either TPICA or PARAFAC no matter how small the tolerance value of the convergence is, how large the maximum

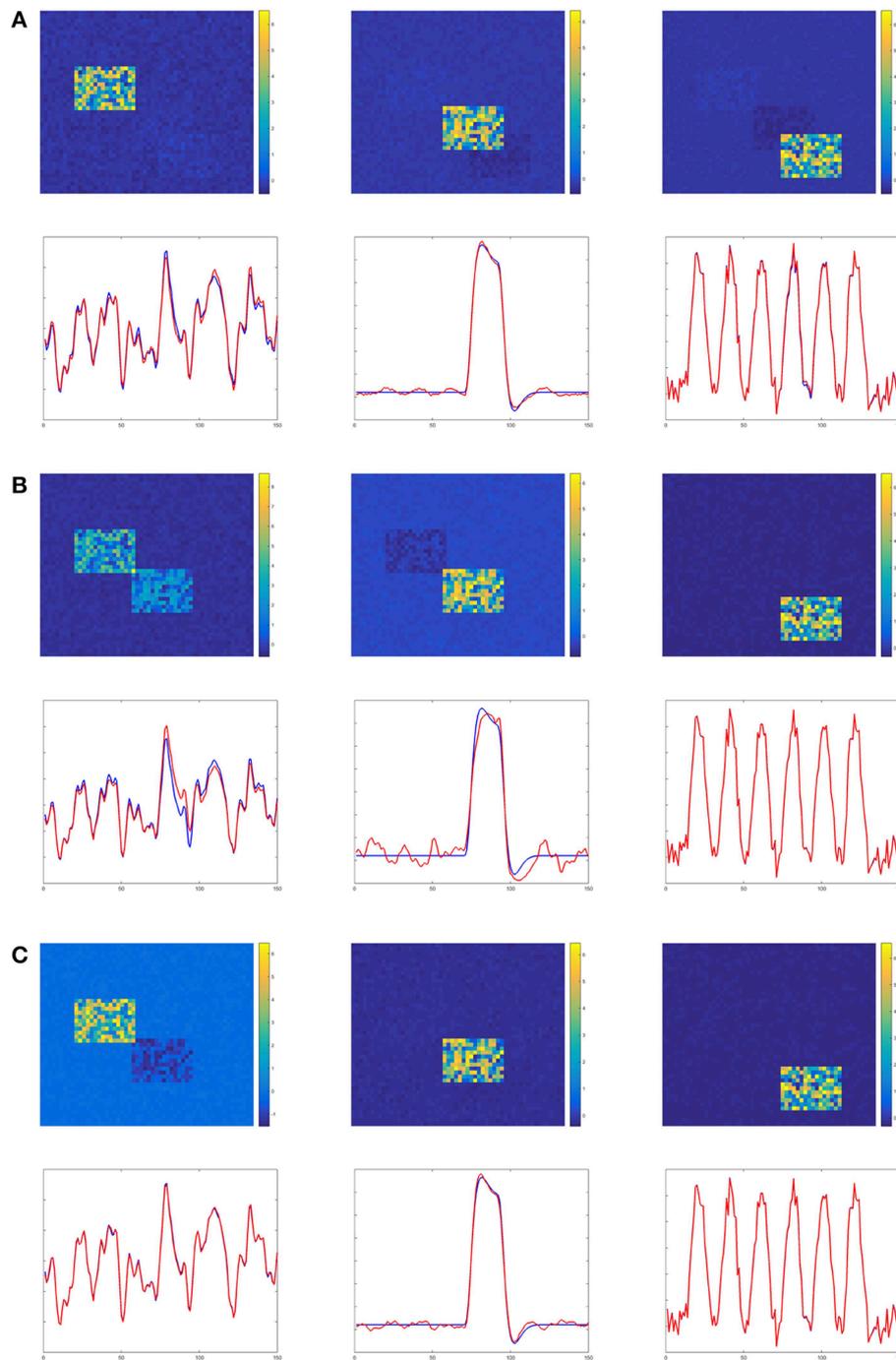


FIGURE 6 | Experiment C in the case of low noise, low overlapping and high collinearity. It shows PARAFAC suffers from high collinearity issue. **(A)** TPICA. **(B)** PARAFAC. **(C)** Proposed Non-Gaussian penalized PARAFAC.

iteration time is, and how exact the initial value is. In other words, the improvement of our proposed algorithm is because of the intrinsic modification of the penalty term and cannot be achieved by parameters setting of the other two algorithms. PARAFAC is not robust even when we use the results of TPICA as the initial value.

However, this proposed algorithm is more time consuming than other algorithms due to the calculations of the penalty term and the tensor product. In the future, we can apply the modified PARAFAC using Hadamard products instead of Khatri-Rao products for large-scale problems to overcome this issue [43, 44]. The spatial smoothness

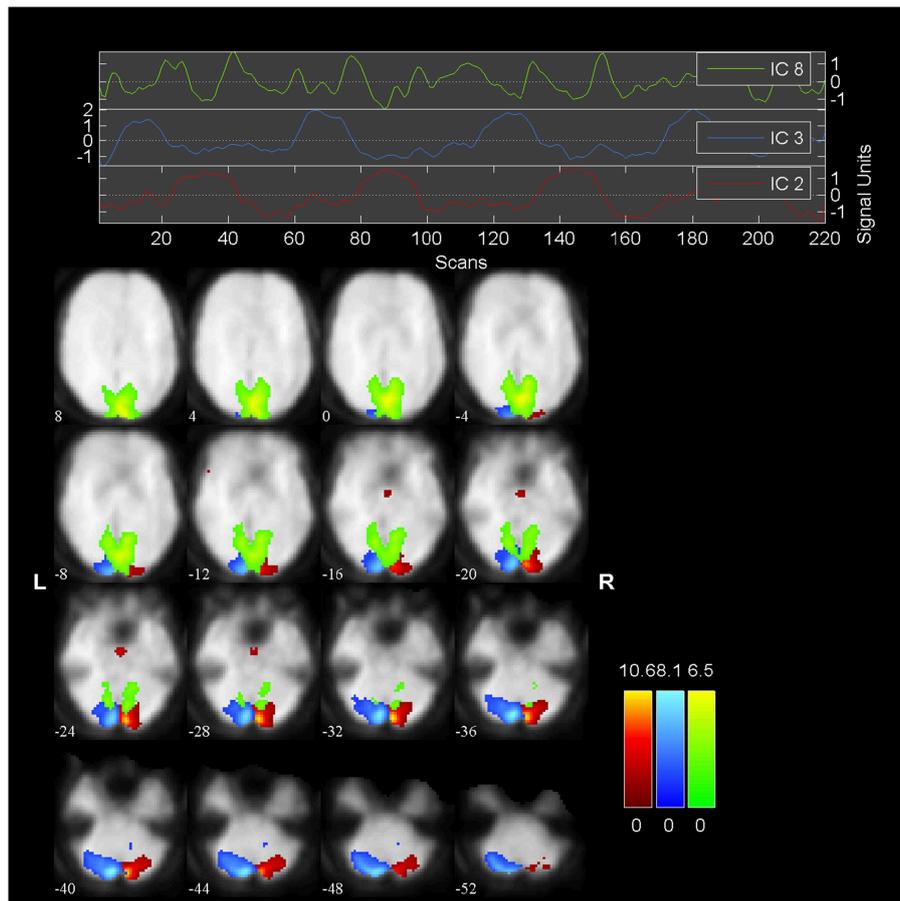


FIGURE 7 | Result of the real fMRI data.

and the dependency between spatial and temporal are not considered in this paper, since these two topics are not the primary concerns in designing this algorithm though they are both very interesting topics and can be a problem for further consideration.

AUTHOR CONTRIBUTIONS

JL, JZ, and DH contributed conception and design of the study. JL performed the statistical analysis and wrote the first draft of the manuscript. JZ and DH improved writing and completed

the manuscript. All authors contributed to manuscript finalizing, read, and approved the submitted version.

ACKNOWLEDGMENTS

The authors are indebted to reviewers for providing insightful and constructive comments and suggestions which has definitely resulted in improvement the readability and quality of the paper. JZ's research was supported by NSFC (No. 61572038) and Chinese State Ministry of Science and Technology (No. 2014ZX02502).

REFERENCES

- Correa N, Adali T, Calhoun VD. Performance of blind source separation algorithms for fMRI analysis using a group ICA method. *Magn Reson Imaging*. (2007) 25:684–94. doi: 10.1016/j.mri.2006.10.017
- Calhoun VD, Liu J, Adali T. A review of group ICA for fMRI data and ICA for joint inference of imaging, genetic, and ERP data. *NeuroImage*. (2009) 45:S163–72. doi: 10.1016/j.neuroimage.2008.10.057
- Allen EA, Erhardt EB, Wei Y, Eichele T, Calhoun VD. Capturing inter-subject variability with group independent component analysis of fMRI data: a simulation study. *NeuroImage*. (2012) 59:4141–59. doi: 10.1016/j.neuroimage.2011.10.010
- Schmithorst VJ, Holland SK. A Comparison of three methods for generating group statistical inferences from independent component analysis of functional magnetic resonance imaging data. *J Magn Reson Imaging*. (2004) 19:365–8. doi: 10.1002/jmri.20009
- Assaf M, Jagannathan K, Calhoun VD, Miller L, Stevens MC, Sahl R, et al. Abnormal functional connectivity of default mode sub-networks in autism spectrum disorder patients. *NeuroImage*. (2010) 53:247–56. doi: 10.1016/j.neuroimage.2010.05.067

6. Lindquist MA. The statistical analysis of fMRI data. *Stat Sci.* (2008) **23**:439–64. doi: 10.1214/09-STS282
7. Hyvärinen A. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans Neural Netw.* (1999) **10**:626–34. doi: 10.1109/72.761722
8. Hyvärinen A. Independent component analysis: recent advances. *Philos Trans A Math Phys Eng Sci.* (2012) **371**:20110534. doi: 10.1098/rsta.2011.0534
9. Lu W, Rajapakse JC. ICA with reference. *Neurocomputing.* (2006) **69**:2244–57. doi: 10.1016/j.neucom.2005.06.021
10. Calhoun VD, Adali T, Stevens MC, Kiehl KA, Pekar JJ. Semiblind ICA of fMRI: a method for utilizing hypothesis-derived time courses in a spatial ICA analysis. *NeuroImage.* (2005) **25**:527–38. doi: 10.1016/j.neuroimage.2004.12.012
11. Lin QH, Liu J, Zheng YR, Liang H, Calhoun VD. Semiblind spatial ICA of fMRI using spatial constraints. *Hum Brain Mapp.* (2010) **31**:1076–88. doi: 10.1002/hbm.20919
12. Lu W, Rajapakse JC. Approach and applications of constrained ICA. *IEEE Trans Neural Netw.* (2005) **16**:203–12. doi: 10.1109/TNN.2004.836795
13. Formisano E, Esposito F, Salle FD, Goebel R. Cortex-based independent component analysis of fMRI time series. *Magn Reson Imaging.* (2004) **22**:1493–504. doi: 10.1016/j.mri.2004.10.020
14. Ma S, Li XL, Correa NM, Adali T, Calhoun VD. Independent subspace analysis with prior information for fMRI data. In: *2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*. Dallas, TX (2010). p. 1922–5.
15. Sharma A, Paliwal KK. Subspace independent component analysis using vector kurtosis. *Patt Recogn.* (2006) **39**:2227–32. doi: 10.1016/j.patcog.2006.04.021
16. Theis FJ. Towards a general independent subspace analysis. In: *Proceedings of the 19th International Conference on Neural Information Processing Systems, NIPS'06*. Vancouver, BC: MIT Press (2006). p. 1361–8.
17. Vollgraf R, Obermayer K. Multi dimensional ICA to separate correlated sources. In: Dietterich TG, Becker S, Ghahramani Z, editors. *Advances in Neural Information Processing Systems 14*. Vancouver, BC: MIT Press (2001). p. 993–1000.
18. Aapo Hyvärinen POH, Inki M. Topographic independent component analysis. *Neural Comput.* (2001) **13**:1527–58. doi: 10.1162/089976601750264992
19. Ge R, Wang Y, Zhang J, Yao L, Zhang H, Long Z. Improved FastICA algorithm in fMRI data analysis using the sparsity property of the sources. *J Neurosci Methods.* (2016) **263**:103–14. doi: 10.1016/j.jneumeth.2016.02.010
20. Hyvärinen A. Independent component analysis in the presence of Gaussian noise by maximizing joint likelihood. *Neurocomputing.* (1998) **22**:49–67. doi: 10.1016/S0925-2312(98)00049-6
21. Hyvärinen A. Fast ICA for noisy data using Gaussian moments. In: *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems (ISCAS 99)*. Vol. 5. Orlando, FL (1999). p. 57–61.
22. Beckmann CF, Smith SM. Probabilistic independent component analysis for functional magnetic resonance imaging. *IEEE Trans Med Imaging.* (2004) **23**:137–52. doi: 10.1109/TMI.2003.822821
23. Tipping ME, Bishop CM. Probabilistic principal component analysis. *J R Stat Soc Ser B.* (1999) **21**:611–22. doi: 10.1111/1467-9868.00196
24. Beckmann CF, Smith SM. Tensorial extensions of independent component analysis for multisubject fMRI analysis. *NeuroImage.* (2005) **25**:294–311. doi: 10.1016/j.neuroimage.2004.10.043
25. Guo Y. A general probabilistic model for group independent component analysis and its estimation methods. *Biometrics.* (2011) **67**:1532–42. doi: 10.1111/j.1541-0420.2011.01601.x
26. Guo Y, Pagnoni G. A unified framework for group independent component analysis for multi-subject fMRI data. *NeuroImage.* (2008) **42**:1078–93. doi: 10.1016/j.neuroimage.2008.05.008
27. Tipping ME, Bishop CM. Mixtures of probabilistic principal component analyzers. *Neural Comput.* (1999) **11**:443–82. doi: 10.1162/089976699300016728
28. Bro R. PARAFAC: tutorial and applications. *Chemometr Intell Lab Syst.* (1997) **38**:149–71. doi: 10.1016/S0169-7439(97)00032-4
29. Harshman RA, Lundy ME. PARAFAC: parallel factor analysis. *Comput Stat Data Anal.* (1994) **18**:39–72. doi: 10.1016/0167-9473(94)90132-5
30. Kolda TG, Bader BW. Tensor decompositions and applications. *SIAM Rev.* (2009) **51**:455–500. doi: 10.1137/07070111X
31. Cichocki A. Tensor decompositions: a new concept in brain data analysis. *CoRR.* (2013) abs/1305.0395.
32. Cichocki A. Era of big data processing: a new approach via tensor networks and tensor decompositions. *CoRR.* (2014) abs/1403.2048.
33. Stegeman A. Comparing independent component analysis and the parafac model for multi-subject artificial fMRI data. In: *Invited Speaker at the IOPS Conference* (2006). Available online at: <http://www.alwinstegeman.nl/docs/Stegeman%20-%20parafac%20&%20ica.pdf>
34. Helwig NE, Hong S. A critique of Tensor Probabilistic Independent Component Analysis: implications and recommendations for multi-subject fMRI data analysis. *J Neurosci Methods.* (2013) **213**:263–73. doi: 10.1016/j.jneumeth.2012.12.009
35. Cao YZ, Chen ZP, Mo CY, Wu HL, Yu RQ. A PARAFAC algorithm using penalty diagonalization error (PDE) for three-way data array resolution. *Analyst.* (2000) **125**:2303–10. doi: 10.1039/b006162j
36. Stegeman A. Using the simultaneous generalized Schur decomposition as a Candecom/Parafac algorithm for ill-conditioned data. *J Chemometr.* (2009) **23**:385–92. doi: 10.1002/cem.1232
37. Vos MD, Nion D, Huffel SV, Lathauwer LD. A combination of parallel factor and independent component analysis. *Signal Process.* (2012) **92**:2990–9. doi: 10.1016/j.sigpro.2012.05.032
38. Eduardo Martínez-Montes JMSB, Valdés-Sosa PA. Penalized parafac analysis of spontaneous EEG recordings. *Stat Sin.* (2008) **18**:1449–64.
39. Martínez-Montes E, Vega-Hernández M, Sánchez-Bornot JM, Valdés-Sosa PA. Identifying complex brain networks using penalized regression methods. *J Biol Phys.* (2008) **34**:315–23. doi: 10.1007/s10867-008-9077-0
40. Martínez-Montes E, Sarmiento-Pérez R, Sánchez-Bornot JM, Valdés-Sosa PA. Information entropy-based penalty for PARAFAC analysis of resting EEG. In: Wang R, Shen E, Gu F, editors. *Advances in Cognitive Neurodynamics ICCN 2007: Proceedings of the International Conference on Cognitive Neurodynamics*. Dordrecht: Springer (2008). p. 443–446.
41. Calhoun VD, Adali T, Pearlson GD, Pekar JJ. A method for making group inferences from functional MRI data using independent component analysis. *Hum Brain Mapp.* (2001) **14**:140–51. doi: 10.1002/hbm.1048
42. Tucker LR. A method for synthesis of factor analysis studies. In: *Technical Report 984*. Princeton, NJ: Educational Testing Service (1951).
43. Phan AH, Tichavsky P, Cichocki A. Fast alternating LS algorithms for high order CANDECOMP/PARAFAC tensor factorizations. *IEEE Trans Sig Process.* (2013) **61**:4834–46. doi: 10.1109/TSP.2013.2269903
44. Phan AH, Cichocki A. PARAFAC algorithms for large-scale problems. *Neurocomputing.* (2011) **74**:1970–84. doi: 10.1016/j.neucom.2010.06.030

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Liang, Zou and Hong. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.