

Analyzing Functional Ultrasound Images using Tensor Decompositions

Shuoyan Zhao (5457602) & Xuan Gao (5232481)

Department of Electrical Engineering

TU Delft

Delft, The Netherlands

S.Zhao-4@student.tudelft.nl & X.Gao-4@student.tudelft.nl

I. PROBLEM FORMULATION

Where and how will the brain respond to a certain stimulus? In order to answer such a fundamental question in neuroscience, functional imaging methods (e.g., functional magnetic resonance imaging (fMRI)) is applied. By imaging hemodynamic changes in brain which are induced by activated neurons, these approaches provide further information about functions of the brain [1]. However, the excellent penetration ability of fMRI comes at the cost of spatial and temporal resolution. In contrast, functional ultrasound (fUS) imaging excels in both penetration depth and resolution [2].

The functional ultrasound data is stored in three-dimensional tensors, which span 2-D space and time. Given a stimulus, brain parts with indeterminate number and location will be activated, which will generate the so-called source signals. These source signals are mixed up and induce fluctuations in the blood dynamics of the entire brain, which will then be observed in imaging measurements. Therefore, to determine which components of the brain are affected by this stimulus, blind source separation is necessary. The goal of blind source separation is to recover the sources purely based on observations. Tensor decomposition techniques are very well suited to solve these problems.

In this assignment, we will implement canonical polyadic decomposition (CPD) and the block-term decomposition (BTD) to separate components from measurements. Both of these two tensor decomposition methods approximate the original tensor with sum of R low-rank tensors, or terms. In another word, they aim to find R independent sources as well as contributions of sources towards observation separately. CPD makes assumptions that components are all rank-one. On the other hand, terms of BTD are generalized to a more common low-rank structure. Moreover, BTD has relatively mild uniqueness conditions compared to CPD. BTD is thus considered a more promising tool for blind source separation [3] [4].

II. CORRELATION IMAGE

The correlation image shown in Fig. 1 maps the Pearson correlation coefficient (PCC) values to the average power-Doppler image (PDI) of the whole test area. The PCC is

computed between the stimulus time-series and the fUS time-series of each corresponding pixel. The brain response is often delayed compared to the stimulus onset. Therefore, the stimulus needs to be delayed at a value that maximizes the average absolute correlation value over the whole image. To emphasize the strong correlation, only PCC larger than 0.3 are displayed.

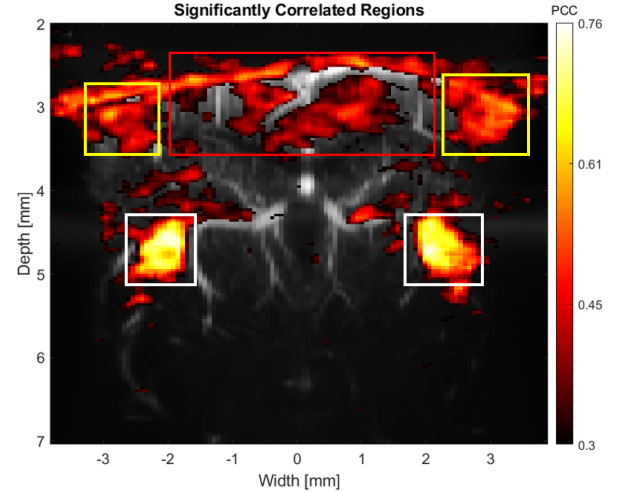


Fig. 1. Image of significant correlation. The PCC is mapped to the average PDI, which is displayed as the black-and-white background. LGN and visual cortex locate at yellow and white colored windows respectively, while the blood vessels are shown in the red-colored window area.

The correlation image indicates the regions that are significantly related to stimulus. Therefore, after tensor decomposition, components whose spatial map points to such areas are expected. In this context, such active areas are lateral geniculate nucleus (LGN), visual cortex, and the large blood vessels on top.

III. APPLYING CPD

In this section, CPD is applied on fUS data. After separation, the extracted component is supposed to appear in the corresponding location of LGN, namely, where yellow spots locate in Fig. 1. Note that during the implementation, the

number of sources, i.e. R , is chosen as 15, which will be justified in section III-D.

A. CPD algorithm

CPD decomposes a tensor of Nth-order into rank-1 terms, which is given by:

$$\begin{aligned}\underline{\mathbf{T}} &\cong \sum_{r=1}^R \lambda_r \mathbf{b}_r^{(1)} \circ \mathbf{b}_r^{(2)} \circ \dots \circ \mathbf{b}_r^{(N)} \\ &= \underline{\mathbf{A}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \dots \times_N \mathbf{B}^{(N)} \\ &= \underline{\mathbf{A}}; \mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \dots, \mathbf{B}^{(N)}\end{aligned}\quad (1)$$

The n-mode matricization $\mathbf{T}_{(n)}$ of the tensor $\underline{\mathbf{T}}$ can be expressed in terms of the Khatri-Rao product of its CPD factor matrices. In a 3-way case, mode-1,2,3 matricization can be expressed as:

$$\begin{aligned}\mathbf{X}_{(1)} &= \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \\ \mathbf{X}_{(2)} &= \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T \\ \mathbf{X}_{(3)} &= \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T\end{aligned}\quad (2)$$

Figure 2 shows the relative iteration error of the CPD.

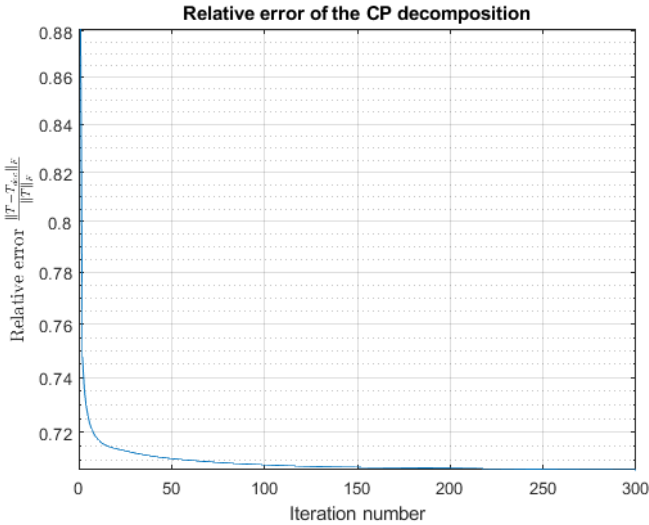


Fig. 2. Relative error plot of the CP decomposition when $R = 15$ and iteration number = 300.

Table I shows the basic alternating least squares (ALS) approach for the CPD of a 3rd-order tensor. ALS algorithm minimizes the cost function iteratively by optimizing each factor matrix, with other component matrices being kept fixed [6].

B. Question 1: Are you able to extract a component whose spatial map points to any of the expected areas?

In Fig. 3, we reconstruct those decomposed rank-one tensors and display their spatial maps. Through comparison between correlation image and the component spatial maps in Fig. 4, it is clear that three of them are meaningful and

TABLE I
ALS ALGORITHM FOR CPD

Algorithm 1 Basic ALS for the CPD of a 3rd-order tensor

Input: $\underline{\mathbf{T}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, and rank R
Output: Factor matrices $\mathbf{A} \in \mathbb{R}^{I_1 \times R}$, $\mathbf{B} \in \mathbb{R}^{I_2 \times R}$, $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$, and scaling vector $\lambda \in \mathbb{R}^R$

- 1: *Initialisation* : $\mathbf{A}, \mathbf{B}, \mathbf{C}$
- 2: **while** not converged or iteration limit is not reached **do**
- 3: $\mathbf{A} \leftarrow \mathbf{T}_{(1)} [(\mathbf{C} \odot \mathbf{B})^T]^\dagger$
- 4: Normalize column vectors of \mathbf{A} to unit length (by computing the norm of each column vector and dividing each element of a vector by its norm)
- 5: $\mathbf{B} \leftarrow \mathbf{T}_{(2)} [(\mathbf{C} \odot \mathbf{A})^T]^\dagger$
- 6: Normalize column vectors of \mathbf{B} to unit length
- 7: $\mathbf{C} \leftarrow \mathbf{T}_{(3)} [(\mathbf{B} \odot \mathbf{A})^T]^\dagger$
- 8: Normalize column vectors of \mathbf{C} to unit length
- 9: Store the norms in vector λ
- 10: **end while**
- 11: **return** : $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and λ

explainable. According to background knowledge mentioned in Fig. 1 and comparison, we consider them as successful extraction of components of LGN, visual cortex and blood vessels, respectively.

C. Question 2: If your answer to the previous question is yes, what information does the temporal signature of that component entail? Is the temporal signature significantly correlated (i.e., with a PCC value above 0.3) to the stimulus?

The third mode of CPD contains the temporal information of every pixel represented by the other two modes. In another word, when a component is extracted, its corresponding third mode will give out information about when such component is active. As shown in sub-titles of Fig. 4, the temporal signatures of the components have significant correlations ($PCC > 0.3$) to the stimulus in the experiment.

D. Question 3: How did you determine the number of sources? Justify your choice.

The selection is based on background information as well as experiment result. To be specific, with different initial conditions (i.e. R value and initialization), we find:

- In our experiment, When R is smaller than 15, the extraction of certain component (such as LGN) can not be guaranteed. When R is larger than 15, although meaningful components are still visible, there are usually some repetitions of components. Example results are shown in Fig. 5 and Fig. 6. In spite of the fact that the result largely depends on random initialisation, we still choose 15 as the optimum.
- Initialization also plays an important role in ALS algorithm. The ALS algorithm will theoretically converge to an optimal point. This optimum, however, is only optimal locally, that is, the best in its small neighborhood. Worse still, such result is usually not

Spatial Maps of CP Decomposed Components

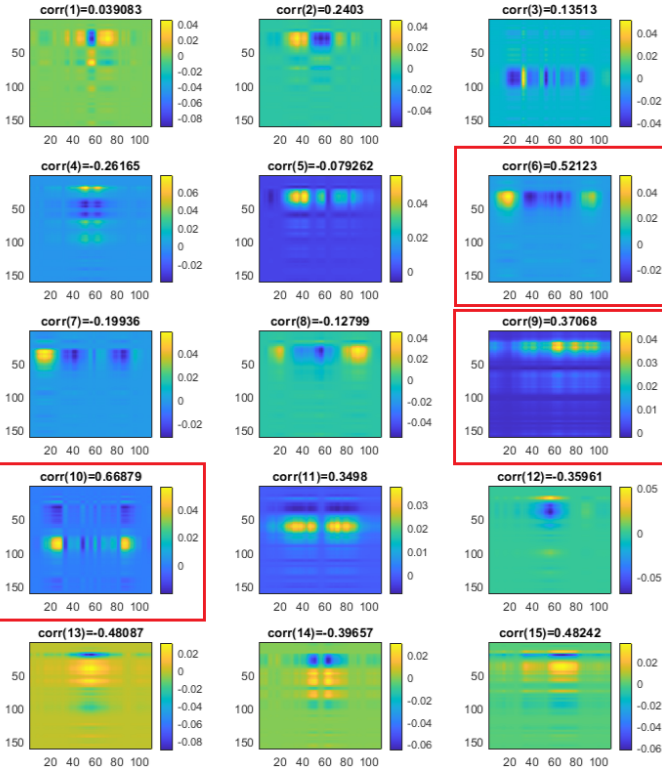


Fig. 3. Spatial maps of CPD components of the $R = 15$. In the sub-titles are the PCC values between the temporal signature of every component and stimulus, respectively. Sub-figure 6, 9 and 10 are selected for further analysis.

Spatial Maps
- comparison between correlation and CPD component

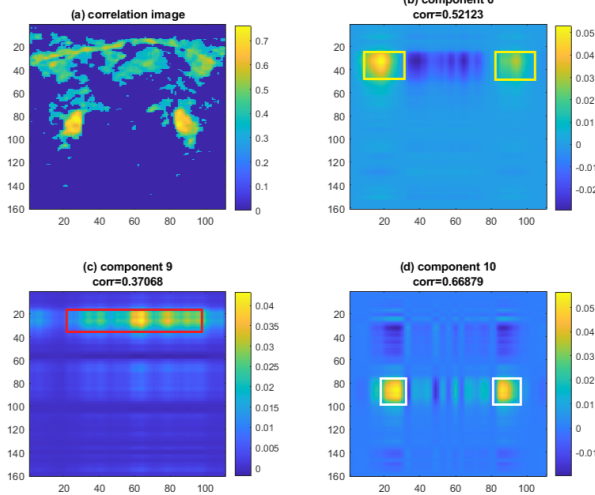


Fig. 4. Comparison between correlation map and spatial maps of CPD components. Subplots are: (a) original correlation map between average DPI and stimulus, (b) extracted component related to visual cortex, (c) component of blood vessels and (d) component of LGN.

Spatial Maps of CP Decomposed Components

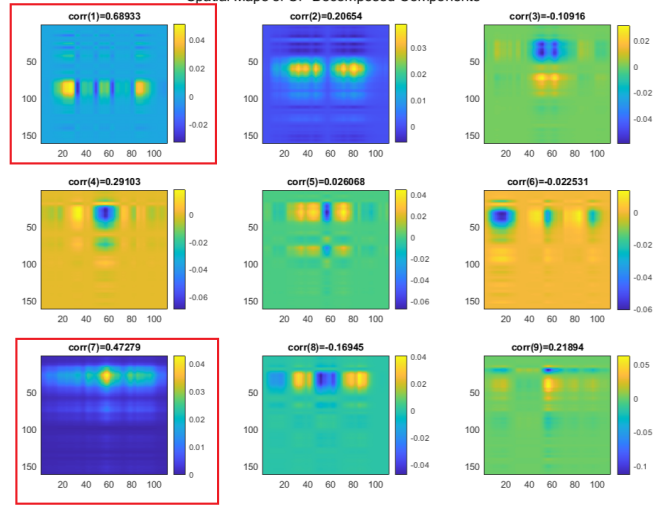


Fig. 5. Spatial maps of CPD components of $R = 9$. The PCC values of the first and seventh components are above 0.3, which indicates a strong correlation of their temporal signature with stimulus. They resemble LGN and blood vessels, while the visual cortex is invisible in such case.

Spatial Maps of CP Decomposed Components

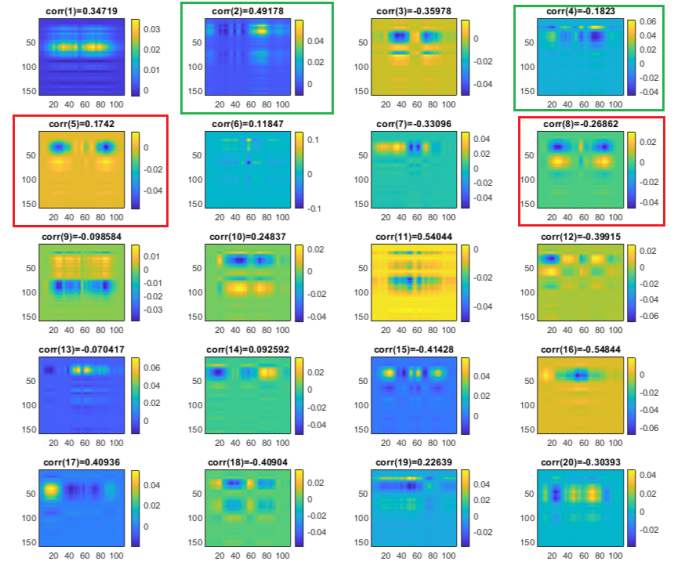


Fig. 6. Spatial maps of CPD components with $R = 20$. There are some images that are duplicated with each other, for example, components 5 and 8, and components 2 and 4.

good enough to decompose meaningful components. Thus trying on several initialization values is necessary. For example, Fig. 7 shows spatial maps of all the $R = 15$ components with different initialization values from Fig. 3. With the same R ($R = 15$) and different initialization values, the extracted components are different. The 6th component in Fig. 7 shows a strong correlation in the visual cortex area. The 8th component in Fig. 7 can also successfully extract the visual cortex activation, while Fig. 3 shows a better result. It is hard to find the perfect visual cortex activation area in Fig. 7. Therefore, the initialization values of Fig. 3 work better than Fig. 7.

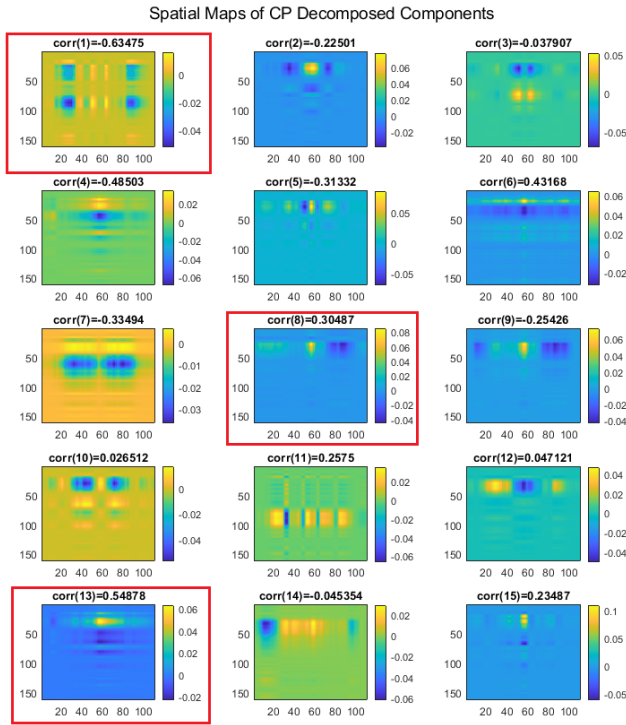


Fig. 7. Spatial maps of CPD components with a different initialization (with $R = 15$ fixed). Note that even the 14th image looks like a cortex area, the PCC is too small to be a meaningful component.

IV. APPLYING $(L_r, L_r, 1)$ -BTD

In this section, BTD is applied on fUS data. Different from CPD which decomposes a tensor into a sum of rank-1 terms, BTD is more flexible by decomposing a tensor into a sum of low multi-linear rank terms [5].

A. BTD algorithm

A third order tensor $\underline{\mathbf{T}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ can be decomposed into the following form:

$$\underline{\mathbf{T}} = \sum_{r=1}^R \mathbf{E}_r \circ \mathbf{c}_r \quad (3)$$

where the matrices \mathbf{E}_r have rank L_r . If the matrices \mathbf{E}_r are factorized into the multiplication of two matrices $\mathbf{A}_r \cdot \mathbf{B}_r^T$ with ranks L_r and $\mathbf{A}_r \in \mathbb{R}^{I_1 \times L_r}$, $\mathbf{B}_r \in \mathbb{R}^{I_2 \times L_r}$, then the tensor BTD can be written as:

$$\underline{\mathbf{T}} = \sum_{r=1}^R (\mathbf{A}_r \cdot \mathbf{B}_r^T) \circ \mathbf{c}_r \quad (4)$$

The factor matrices can be written as:

$$\mathbf{A} = [\mathbf{A}_1 \dots \mathbf{A}_R] \text{ and } \mathbf{B} = [\mathbf{B}_1 \dots \mathbf{B}_R] \text{ and } \mathbf{C} = [\mathbf{c}_1 \dots \mathbf{c}_R]$$

BTD in matrix forms can be written as:

$$\begin{aligned} \mathbf{T}_{(1)} &= \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \\ \mathbf{T}_{(2)} &= \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T \\ \mathbf{T}_{(3)} &= \mathbf{C}[(\mathbf{B}_1 \odot \mathbf{A}_1) \mathbf{1}_{L_r} \dots (\mathbf{B}_R \odot \mathbf{A}_R) \mathbf{1}_{L_r}]^T \end{aligned} \quad (5)$$

where $\mathbf{T}_{(n)}$ is the mode- n matricization of tensor $\underline{\mathbf{T}}$, \odot is the Khatri-Rao product, and $\mathbf{1}_{L_r}$ is a vector of L_r size. As $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$, while $\mathbf{B} \in \mathbb{R}^{I_2 \times RL_r}$, $\mathbf{C} \odot \mathbf{B}$ can be calculated as:

$$(\mathbf{C} \odot \mathbf{B})_{I_2 I_3 \times RL} = [(\mathbf{C}_1 \otimes \mathbf{B}_1)_{I_2 I_3 \times L} \dots (\mathbf{C}_R \otimes \mathbf{B}_R)_{I_2 I_3 \times L}]$$

The same method applies to the calculation of $\mathbf{C} \odot \mathbf{A}$:

$$(\mathbf{C} \odot \mathbf{A})_{I_1 I_2 \times RL} = [(\mathbf{C}_1 \otimes \mathbf{A}_1)_{I_1 I_2 \times L} \dots (\mathbf{C}_R \otimes \mathbf{A}_R)_{I_1 I_2 \times L}]$$

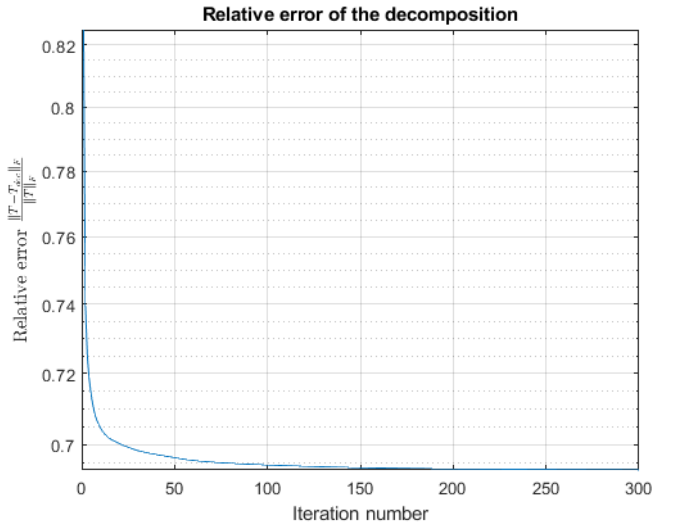


Fig. 8. Relative error plot of BTD when $R = 12$, $L = 2$, and iteration number = 300.

Table II shows the basic ALS for the BTD of a 3rd-order tensor.

TABLE II
ALS ALGORITHM OF $(L_r, L_r, 1)$ -BTD

Algorithm 2 Basic ALS for the BTD of a 3rd-order tensor (1)

Input: $\mathbf{T} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, rank R , and rank L_r
Output: Factor matrices $\mathbf{A} \in \mathbb{R}^{I_1 \times RL_r}$, $\mathbf{B} \in \mathbb{R}^{I_2 \times RL_r}$, $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$, and scaling vector $\lambda \in \mathbb{R}^R$

- 1: *Initialisation* : $\mathbf{A}, \mathbf{B}, \mathbf{C}$
- 2: **while** not converged or iteration limit is not reached **do**
- 3: $\mathbf{A} \leftarrow \mathbf{T}_{(1)} [(\mathbf{C} \odot \mathbf{B})^T]^\dagger$
- 4: $\mathbf{B} \leftarrow \mathbf{T}_{(2)} [(\mathbf{C} \odot \mathbf{A})^T]^\dagger$
- 5: Normalize column vectors of \mathbf{B} to unit length (by computing the norm of each column vector and dividing each element of a vector by its norm)
- 6: $\mathbf{C} \leftarrow \mathbf{T}_{(3)} [(\mathbf{B}_1 \odot \mathbf{A}_1) \mathbf{1}_{L_r} \dots (\mathbf{B}_R \odot \mathbf{A}_R) \mathbf{1}_{L_r}]^T$
- 7: Normalize column vectors of \mathbf{C} to unit length
- 8: Store the norms in vector λ
- 9: **end while**
- 10: **return** : $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and λ

B. Question 1: Are you able to extract a component whose spatial map points to any of the expected areas? What information does the temporal signature of that component entail? Is the temporal signature significantly correlated to the stimulus?

In Fig. 9, we reconstruct components and display their spatial maps, where we assume that the $R = 12$ sources all have a $rank = 2$ structure. This time, through comparison between correlation image and the component spatial maps in Fig. 10, we can still find our expected components. Similarly as result in CPD, temporal signature of components are highly correlated to the stimulus.

For the initialization selection, we run the algorithm for a small number of iterations (50) first. We select the solution which gives the best fit and save the current initialization. Then we run the algorithm with the saved initialization for the maximum iteration numbers (300) to do further result analysis.

C. Question 2: While applying BTD, which mode of the fUS tensor did you select to be rank-1 and why?

The third mode, i.e. the mode which stands for the time samples of the experiment, is selected to be $rank = 1$. The two spatial factors (width and depth) are of low rank L , while the time mode is of rank one. The selection is made based on the fact that samples are independent on each other.

D. Question 3: Compared to the CPD, what kind of differences do you observe in the extracted components of interest.

Compared to CPD, each component extracted by BTD seems to display information for more than one brain slice, that is, multiple regions in spatial maps. Such regions are similar to the ones in CPD maps, which are, however, displayed separately in each component of CPD. For example, Fig. 11 shows the comparison of spatial maps of CPD and BTD with LGN extraction. The spatial map of CPD only displays one pattern in which the shape of the upper blue area

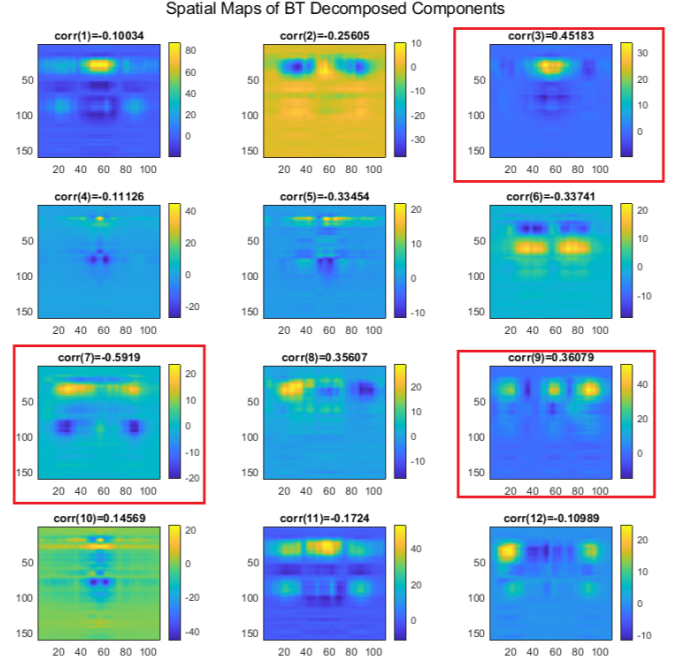


Fig. 9. Spatial maps of all components decomposed by a $(2, 2, 1)$ -BTD of $R = 12$. Subfigure 3, 7 and 9 are selected for further analysis.

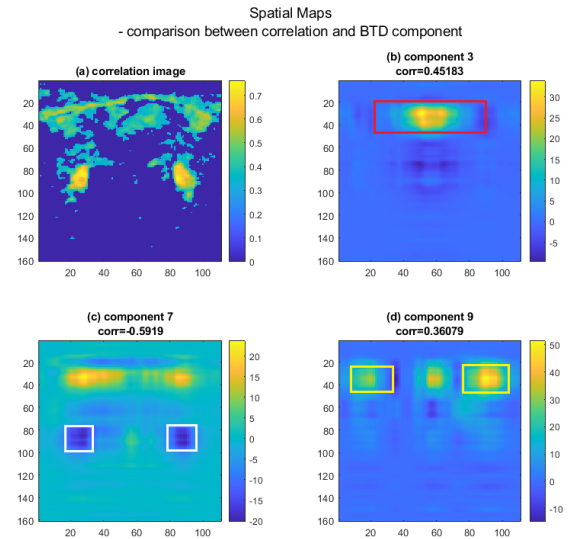


Fig. 10. Comparison between correlation map and component map

looks the same as the lower yellow area. The information in this CPD map is repeated longitudinally. In BTd spatial map, there are two different patterns displayed. The massive blue area and the middle yellow area. It makes sense as the spatial rank $L = 2$.

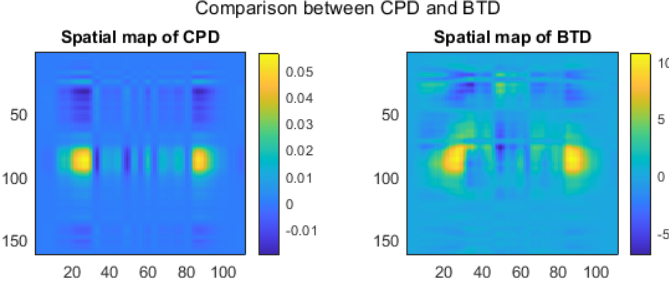


Fig. 11. Comparison between spatial maps of CPD and BTd. Information in CPD maps are repeated longitudinally and the rank is one. BTd map, on the other hand, has different top and middle shape. There are two patterns in this BTd component.

E. Question 4: what do you think is the reason for your observations in the previous question? (Hint: Think of the assumptions made by the CPD and the BTd)

In CPD the components are assumed to be all $rank = 1$, while in BTd to be $rank = L$. Therefore, only one pattern can be observed in CPD components, while L patterns can be observed in BTd components. For example, in Fig. 3, there are several strongly correlated areas in one component, but they have the same pattern with different scaling. In Fig. 9 with $L = 2$, two different patterns can be observed.

F. Question 5: How did you determine the number of sources and the factor matrices rank L ?

- With different number of sources, the situation is similar as the case CPD: less sources sometimes leads to no meaningful components and more sources may be redundant.
- L is connected to the "spatial rank" of every component. BTd will degrade to CPD when $L = 1$. Meanwhile, Fig. 12 shows an example of BTd with $R = 12$ and $L = 3$. The spatial maps are somehow "disordered". One reason of that is more noise is considered to as a part of expected information due to their multiple rank assumption. Besides that "distortion", LGN, visual cortex, and blood vessels are still visible in components 4, 6, and 9 respectively. When L is overestimated and higher than the actual "spatial rank", some pairs of the first two modes does not contain signal information. BTd is relatively insensitive to overestimation of L [8]. In addition, higher L values result in increased complexity. Overestimating L brings in more useless information also increases complexity. Therefore, selecting an appropriate L value is necessary.

Spatial Maps of BT Decomposed Components

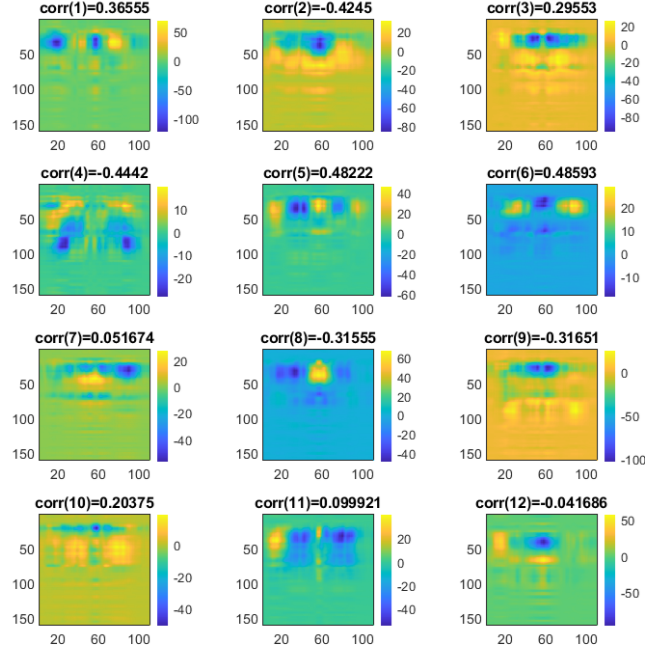


Fig. 12. Spatial maps of all components produced by a $(3, 3, 1)$ -BTd with $R = 12$. LGN, visual cortex, and blood vessel components are still visible.

- Due to using of ALS algorithm, our statement of role of initialization in BTd is same as before. Fig. 13 shows an example, starting at different initialization values from Fig. 9 but with $R = 12$ and $L = 2$ fixed. The extracted components are obviously different. The 1st component in Fig. 9 shows a strong correlation in the blood vessel area. The 3rd component in Fig. 13 can also successfully extract the blood vessel activation, but in a worse quality. Overall, Fig. 9 work better than Fig. 13.

V. ADDITIONAL DISCUSSIONS

Among different tensor decomposition methods, CPD, and BTd can both be considered as constrained Tucker decomposition. The core tensor of CPD only has values on the super-diagonal, while there can be non-zero values on the off-diagonal for the core tensor of BTd. The specific positions of non-zero entries depend on the multilinear rank of the factor matrices. When the spatial rank L of BTd is one, it is the same as CPD. BTd is a generalization of CPD.

BTd is more flexible that reveals the low rankness of the spatial mode compared to CPD, as CPD is only applicable for one-way data while BTd is appropriate for multi-way data. As some physiological artifacts are not likely to satisfy rank-1 assumption [8], hence using BTd can better adapt to those models.

From our results, it is also observable that different R values (overestimation and underestimation) can easily affect

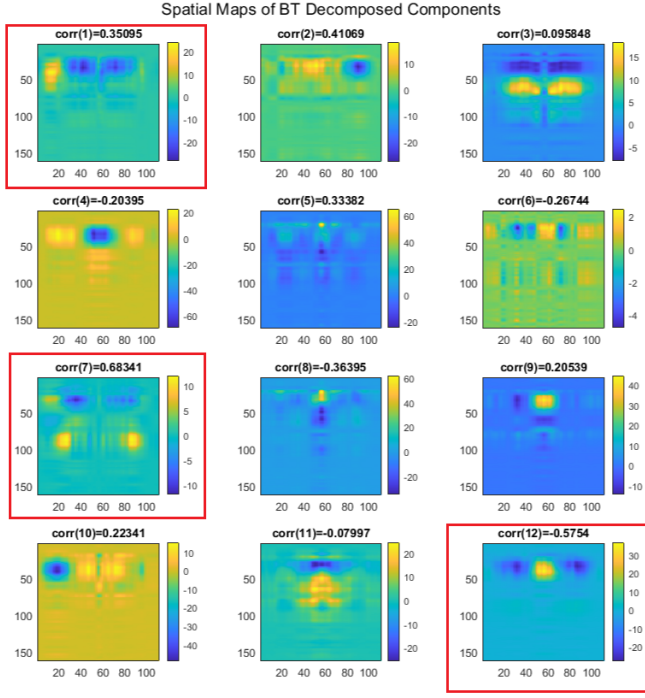


Fig. 13. Spatial maps of BT components with a different initialization (with $R = 12$ fixed). In red rectangle marked the proper meaningful components.

the effectiveness of the result and cause loss of accuracy. Both CPD and BT are sensitive to the correct estimation of R . For BT, when different L values ($L = 2$ and $L = 3$) are applied, key information can be extracted in both cases. The difference is that there are more phantoms in each component when $L = 3$.

Besides algorithm 2, there is another algorithm (Algorithm 3) to calculate the basic ALS for the BT of a 3rd-order tensor with an extra QR factorization step for \mathbf{B} [7]. The QR decomposition of \mathbf{B} is given by the matrix product \mathbf{QR} , with \mathbf{Q} having orthonormal columns. Fig. 14 shows the relative error using algorithm 3. With an extra QR factorization being added, the convergence does not always apply. After 300 iterations, the relative error is around 0.7, which is the same as the convergence value in Fig. 8 using algorithm 2. Fig. 15 and Fig. 16 show the spatial maps of components with $R = 12$ and $L = 2$ using algorithm 3. The same initialization is applied to algorithm 2 and 3. Compared to algorithm 2 results in Fig. 9, the blood vessel, visual cortex, and LGN in Fig. 16 can also be extracted successfully. The results using algorithm 2 turn out slightly better.

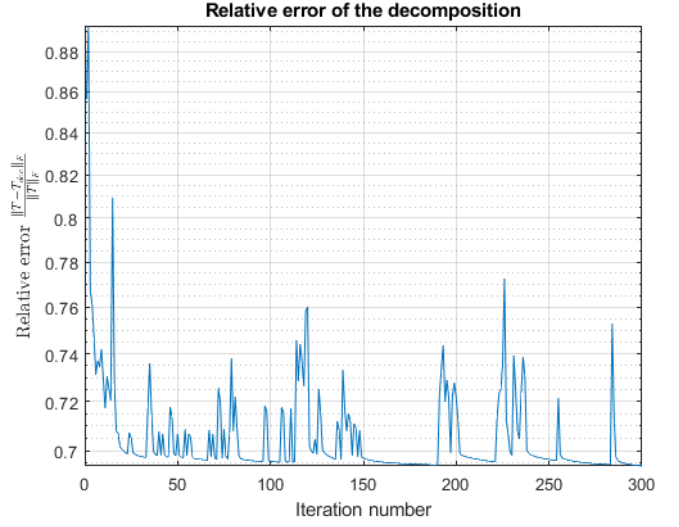


Fig. 14. Relative error plot of BT using algorithm 3 when $R = 12$, $L = 2$, and iteration number = 300.

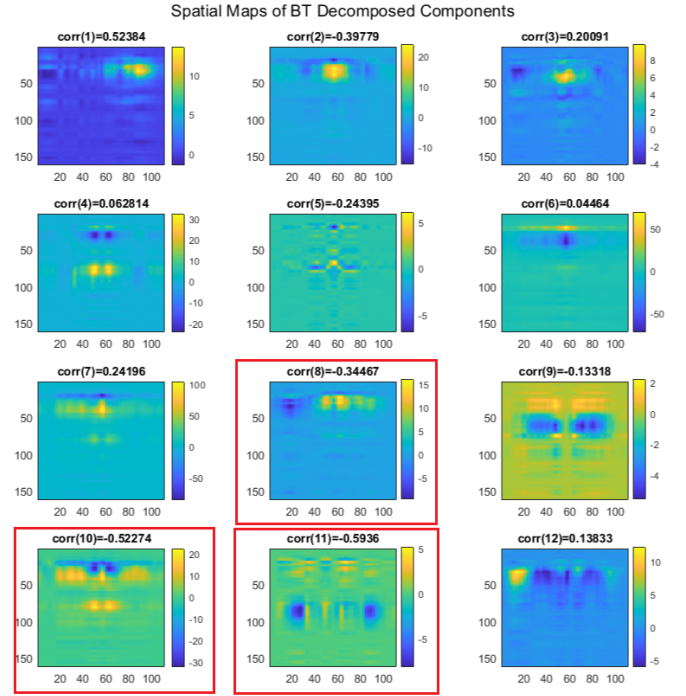


Fig. 15. Spatial maps of all the $R = 12$ and $L = 2$ components using algorithm 3. Subfigure 8, 10 and 11 are selected for further analysis.

TABLE III
ALS ALGORITHM WITH QR FACTORIZATION OF $(L_r, L_r, 1)$ -BTD

Algorithm 3 Basic ALS for the BTD of a 3rd-order tensor (2)

Input: $\mathbf{T} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, rank R , and rank L_r
Output: Factor matrices $\mathbf{A} \in \mathbb{R}^{I_1 \times RL_r}$, $\mathbf{B} \in \mathbb{R}^{I_2 \times RL_r}$, $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$, and scaling vector $\lambda \in \mathbb{R}^R$

- 1: *Initialisation* : $\mathbf{A}, \mathbf{B}, \mathbf{C}$
- 2: **while** not converged or iteration limit is not reached **do**
- 3: $\mathbf{A} \leftarrow \mathbf{T}_{(1)} [(\mathbf{C} \odot \mathbf{B})^T]^\dagger$
- 4: $\tilde{\mathbf{B}} \leftarrow \mathbf{T}_{(2)} [(\mathbf{C} \odot \mathbf{A})^T]^\dagger$
- 5: Normalize column vectors of $\tilde{\mathbf{B}}$ to unit length (by computing the norm of each column vector and dividing each element of a vector by its norm)
- 6: QR factorization: $\tilde{\mathbf{B}}_r = \mathbf{Q}\mathbf{R}$, $\mathbf{B}_r \leftarrow \mathbf{Q}$
- 7: $\mathbf{C} \leftarrow \mathbf{T}_{(3)} [(\mathbf{B}_1 \odot \mathbf{A}_1) \mathbf{1}_{L_r} \dots (\mathbf{B}_R \odot \mathbf{A}_R) \mathbf{1}_{L_r}]^T]^\dagger$
- 8: Normalize column vectors of \mathbf{C} to unit length
- 9: Store the norms in vector λ
- 10: **end while**
- 11: **return** : $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and λ

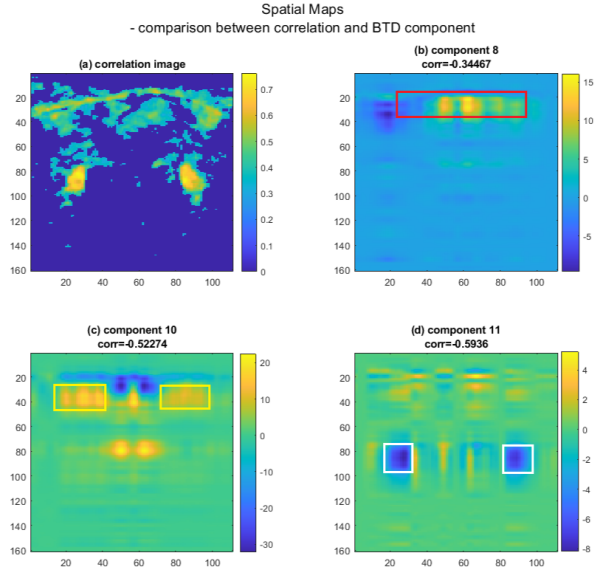


Fig. 16. Comparison between correlation map and component map using algorithm 3. In rectangle marked the proper meaningful components.

REFERENCES

- [1] E. Mace, G. Montaldo, B. -F. Osmanski, I. Cohen, M. Fink and M. Tanter, "Functional ultrasound imaging of the brain: theory and basic principles," in IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 60, no. 3, pp. 492-506, March 2013
- [2] E. Mace, G. Montaldo, I. Cohen, M. Baulac, M. Fink, and M. Tanter, "Functional ultrasound imaging of the brain," Nature Methods, vol. 8, pp. 662-664, 2011
- [3] L. De Lathauwer, "Blind separation of exponential polynomials and the decomposition of a tensor in rank- $(l_r, l_r, 1)$ terms," SIAM Journal on Matrix Analysis and Applications, vol. 32, no. 4, pp. 1451-1474, 2011.
- [4] L. D. Lathauwer, "Block component analysis, a new concept for blind source separation," in International Conference on Latent Variable Analysis and Signal Separation, pp. 1-8, Springer, 2012.
- [5] B. Hunyadi, D. Camps, L. Sorber, W. V. Paesschen, M. D. Vos, S. V. Huffel, and L. D. Lathauwer, "Block term decomposition for modelling

epileptic seizures," EURASIP Journal on Advances in Signal Processing, vol. 2014, no. 1, 2014.

- [6] R.A. Harshman. Foundations of the PARAFAC procedure: models and conditions for an explanatory multimodal factor analysis. UCLA Working Papers in Phonetics, 16:1-84, 1970.
- [7] L. De Lathauwer and D. Nion, "Decompositions of a higher-order tensor in block terms—part III: Alternating least squares algorithms," SIAM Journal on Matrix Analysis and Applications, vol. 30, no. 3, pp. 1067-1083, 2008.
- [8] C. Chatzichristos, E. Kofidis, M. Morante, and S. Theodoridis, "Blind fmri source unmixing via higher-order tensor decompositions," Journal of Neuroscience Methods, vol. 315, pp. 17-47, 2019.