

Analyzing Functional Ultrasound Images using Tensor Decompositions

Yongsheng Han, 5763371

Botao Sun, 5759609

code link: https://github.com/blacksilvergeek/tensor_fus

I. PROBLEM FORMULATION

The objective of this study is to employ tensor decomposition techniques to analyze functional ultrasound (fUS) data, with the aim of capturing the brain's response to sensory stimuli. Functional ultrasound is an innovative neuroimaging technique that measures local blood flow dynamics as a result of neural activity. Our primary focus will be on Canonical Polyadic Decomposition (CPD) and Block Term Decomposition (BTD) to uncover brain regions associated with stimulus tasks.

A. Define the Problem

Functional ultrasound (fUS) data is commonly structured as a third-order tensor, with the first two modes denoting the spatial dimensions of power-Doppler images (PDIs), while the last mode signifies the temporal dimension. To observe the correlation between stimuli and brain region localization, it is essential to separate the third mode with the first two modes. The challenge we are addressing is the application of tensor decomposition techniques to analyze functional ultrasound data, thereby capturing the brain's response to sensory stimuli. The methodologies under investigation include Canonical Polyadic Decomposition (CPD) and Block Term Decomposition (BTD).

B. Research Questions

This assignment aims to investigate several questions related to the tensor decomposition of fUS data. Our objectives include assessing the efficacy of BTD and CPD in extracting components associated with stimulated brain regions, exploring strategies for source number and factor matrix rank determination, comparing the pros and cons of CPD and BTD, and conducting a thorough analysis of the reasons. Additionally, we aim to propose an enhanced approach for handling fUS data.

II. CORRELATION IMAGE

As a reference, Pearson correlation coefficient (PCC) values between the stimulus time-series and the fUS time-series of each corresponding pixel are displayed on the average power-Doppler images (PDIs) first. Since the brain response to a given stimulus is often delayed compared to the stimulus onset, to maximize the average absolute correlation value, we choose the delayed time to be 1.47 seconds. Figure 1 displays

the correlation image where the blue parts represent lateral geniculate nucleus (LGN), the green parts represent visual cortex, and the yellow part represents blood vessels. These regions are significantly related to the stimulus which will be compared with the CPD and BTD results in the following parts.

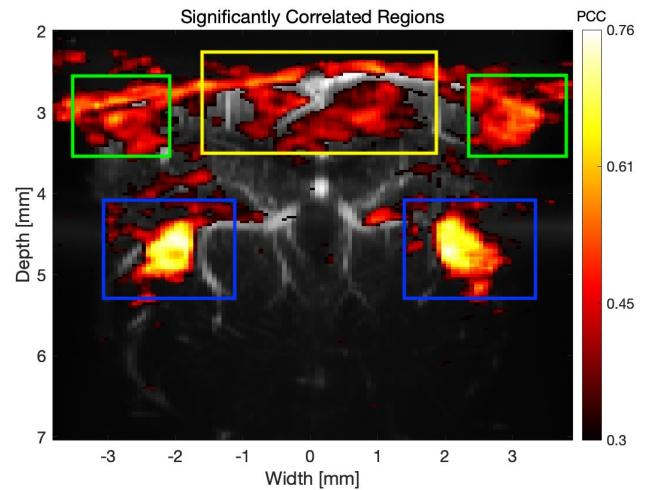


Fig. 1. Correlation image between the stimulus time-series and the fUS time-series of each corresponding pixel

III. PRINCIPLE OF CPD

Given a tensor $\underline{\mathbf{T}}$ of order N , the CPD algorithm decomposes it into a sum of R rank-1 tensors, as shown in the equation:

$$\underline{\mathbf{T}} \cong \sum_{r=1}^R \lambda_r \mathbf{b}_r^{(1)} \circ \mathbf{b}_r^{(2)} \circ \cdots \circ \mathbf{b}_r^{(N)}$$

Here, λ_r are the scaling factors, $\mathbf{b}_r^{(n)}$ are the factor matrices, and \circ denotes the outer product. The decomposition can also be expressed in terms of the mode-n matricization of the tensor $\underline{\mathbf{T}}$, which is denoted as $\mathbf{T}_{(n)}$.

In the case of a 3-way tensor, the mode-1, mode-2, and mode-3 matricizations can be expressed as:

$$\mathbf{X}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T$$

$$\mathbf{X}_{(2)} = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T$$

$$\mathbf{X}_{(3)} = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$$

Here, \odot denotes the Khatri-Rao product, which is a column-wise Kronecker product. The factor matrices A , B , and C correspond to the different modes of the tensor.

IV. CPD RELATED QUESTIONS

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

To enhance the clarity of the results, we have modified the original function `display_brain_img` and introduced a new function named `display_brain_img_sub`. This new function incorporates two significant improvements:

- 1) The alpha channel of the background image is set to 0.4, ensuring that the background remains visible while highlighting the foreground image, thereby improving the overall readability of the image.
- 2) To address the issue where the color black, representing negative correlations, might blend in with the background color, we first identify the sign of the location with the maximum absolute value of correlation. Based on this sign, we then decide whether to invert the correlation image.

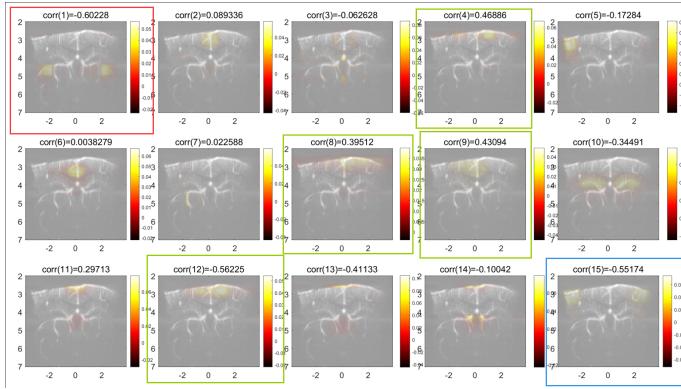


Fig. 2. Result of CPD

Yes, we are able to extract a component whose spatial map points to any of the expected areas. In Figure 2, the red box shows the LGN, and the blue box corresponds to the visual cortex. However, for blood vessels, there is no unique result corresponding to them. As shown in the figure, both green boxes can represent blood vessels. That is, in our experiments, we could not represent this part with a single component.

Question IV-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?

Since we have decomposed the tensor into R three-way tensors, the multiplication of the signatures from the first two dimensions yields a spatial map that identifies active regions. Meanwhile, the feature from the last dimension, the temporal signature, represents the dynamic activity of these regions over time.

As annotated in Figure 2, the correlations for LGN and visual cortex are 0.60 and 0.55, respectively, both of which are greater than 0.3. Similarly, for blood vessels, the correlation is also greater than 0.3.

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

The selection is based on background information as well as experimental results. Specifically, with different initial conditions (i.e., R value and initialization), we find:

- 1) In our experiment, when R is smaller than 15, the extraction of certain components (such as LGN) cannot be guaranteed. When R is 12, as shown in Figure 4, certain components are not extracted properly. On the other hand, when R is larger than 15, although meaningful components are still visible, there are usually some repetitions of components. When R is 20, as shown in Figure 3, repetitions of components can be observed. Despite the fact that the result largely depends on random initialization, we still choose 15 as the optimum.
- 2) Initialization also plays an important role in the ALS algorithm. The ALS algorithm will theoretically converge to an optimal point. However, this optimum is only locally optimal, meaning it is the best in its small neighborhood. Unfortunately, such a result is usually not good enough to decompose meaningful components. Therefore, trying several initialization values is necessary.

By the way, the rank 15 is not truly optimal because the decomposition obtained by this method varies with each iteration, requiring numerous experiments. Similar results can be achieved with ranks close to 15, indicating a degree of robustness in the choice of rank for the decomposition.

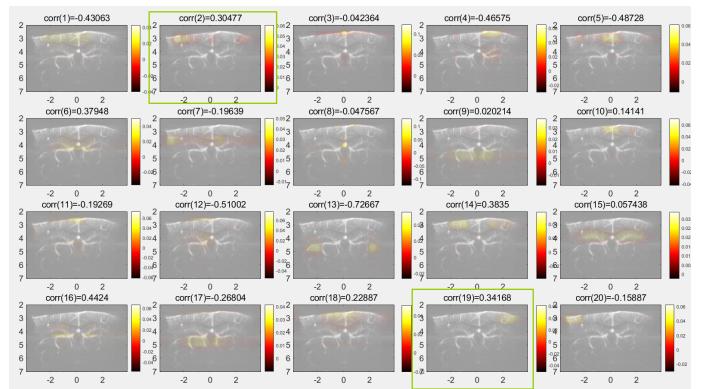


Fig. 3. Result of CPD at R=20

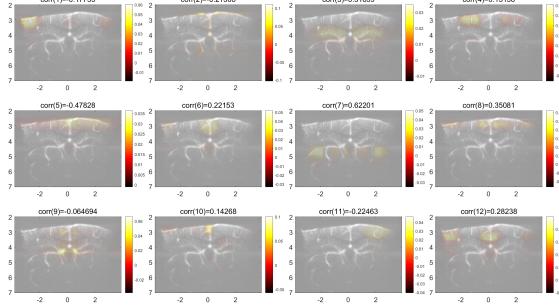


Fig. 4. Result of CPD at $R=12$

V. PRINCIPLE OF $(L_r, L_r, 1)$ -BTD

In this assignment, we aim to perform BTD on three-order fUS data, represented as a tensor $\underline{\mathbf{T}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. $\underline{\mathbf{T}}$ can be represented as a decomposition in the form:

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

where the matrices \mathbf{E}_r are rank L_r , $1 \leq r \leq R$. If the matrix \mathbf{E}_r is factorized to $\mathbf{A}_r \cdot \mathbf{B}_r^T$, with $\mathbf{A}_r \in \mathbb{R}^{I_1 \times L_r}$ and $\mathbf{B}_r \in \mathbb{R}^{I_2 \times L_r}$, then tensor $\underline{\mathbf{T}}$ can be written as:

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

the factor matrices are listed in the following: $\mathbf{A} = [\mathbf{A}_1 \dots \mathbf{A}_R]$ and $\mathbf{B} = [\mathbf{B}_1 \dots \mathbf{B}_R]$ and $\mathbf{C} = [\mathbf{c}_1 \dots \mathbf{c}_R]$.

Then, we can write $BTD - (L_r, L_r, 1)$ as:

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

Where \odot_p is the partition-wise Kronecker product. Furthermore, $\mathbf{A} \odot_p \mathbf{B} = (\mathbf{A}_1 \otimes \mathbf{B}_1 \dots \mathbf{A}_R \otimes \mathbf{B}_R)[2]$. $\mathbf{X}_{(1)}$ and $\mathbf{X}_{(2)}$ can be represented as:

$$\begin{aligned} \mathbf{X}_{(1)} &= \mathbf{A} [(\mathbf{C}_1 \otimes \mathbf{B}_1)_{I_2 I_3 \times L} \dots (\mathbf{C}_R \otimes \mathbf{B}_R)_{I_2 I_3 \times L}] \\ \mathbf{X}_{(2)} &= \mathbf{B} [(\mathbf{C}_1 \otimes \mathbf{A}_1)_{I_1 I_3 \times L} \dots (\mathbf{C}_R \otimes \mathbf{A}_R)_{I_1 I_3 \times L}] \end{aligned}$$

The updated results of factor matrices can be calculated by multiplying mode-n matricization of tensor with pseudo-inverse of the partition-wise Kronecker product.

The following table displays the whole process of implementing $BTD - (L_r, L_r, 1)$ algorithm:

Algorithm 1 $BTD - (L_r, L_r, 1)$ algorithm

Require: $\underline{\mathbf{T}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, rank R and L_r

- 1: Initializing $\mathbf{A} \in \mathbb{R}^{I_1 \times RL_r}$, $\mathbf{B} \in \mathbb{R}^{I_2 \times RL_r}$, $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$ and normalizing by columns
 - 2: **While** not converged or iteration limit is not reached do
 - 3: Calculating the mode-n matricization of tensor \mathbf{T} as $\mathbf{T}_{(1)}$, $\mathbf{T}_{(2)}$ and $\mathbf{T}_{(3)}$
 - 4: $\mathbf{A} \leftarrow \mathbf{T}_{(1)} [(\mathbf{C} \odot_p \mathbf{B})^T]^\dagger$
 - 5: Normalizing \mathbf{A} by columns
 - 6: $\mathbf{B} \leftarrow \mathbf{T}_{(1)} [(\mathbf{C} \odot_p \mathbf{A})^T]^\dagger$
 - 7: Normalizing \mathbf{B} by columns
 - 8: $\mathbf{C} \leftarrow \mathbf{T}_{(3)} [((\mathbf{B}_1 \odot \mathbf{A}_1) \mathbf{1}_{L_r} \dots (\mathbf{B}_R \odot \mathbf{A}_R) \mathbf{1}_{L_r})^T]^\dagger$
 - 9: Store the norms in vector λ
 - 10: Normalizing \mathbf{C} by columns
 - 11: Calculating relative error
 - 12: **End while**
 - 13: **return** 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 $\lambda \in \mathbb{R}^R$
-

VI. BTD RELATED QUESTIONS

Question VI-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 (i.e., with a PCC value above 0.3) to the stimulus?

Yes, we are able to extract a component whose spatial map points to the expected areas. Figure 5 and Figure 6 display PDI spatial maps resulting from BTD with $L = 2$ and $L = 3$, both with $R = 12$. Pearson correlation coefficients between the temporal signature and the stimulus are annotated on each sub-figure.

For $L = 2$, component 10 corresponds to the visual cortex, the LGN, and component 6, 9 corresponds to the blood vessels area. The respective Pearson correlation coefficients are -0.55, 0.43 and -0.31, both exceeding an absolute value of 0.3. The negative values are due to sign ambiguity in the BTD algorithm. Similarly as using CPD algorithm, temporal signature of components are highly correlated to the stimulus.

For $L = 3$, component 9 corresponds to the blood vessels area, while other components represent a combination of expected and unexpected areas. However, the temporal signature still exhibits a high PCC above 0.3 relative to the stimulus. This observation suggests that $L = 2$ is a more suitable choice than $L = 3$ when $R = 12$. We will further explore this in subsequent discussions.

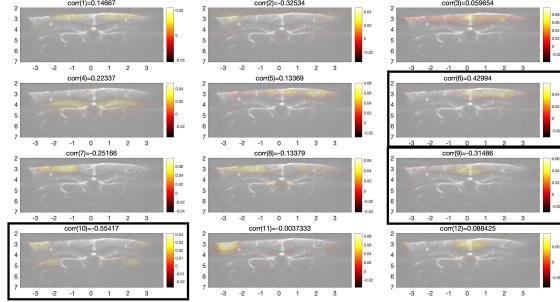


Fig. 5. Spatial maps of BTD when $L = 2$

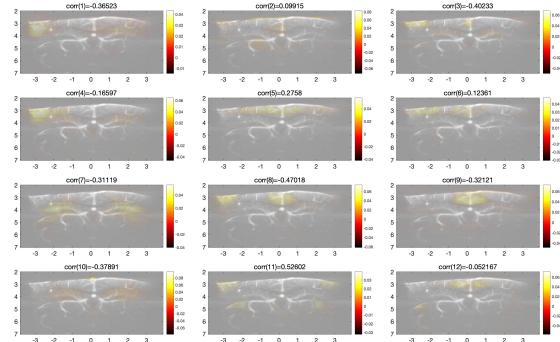


Fig. 6. Spatial maps of BTD when $L = 3$

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

The third mode of the fUS tensor represents the time information was selected to be rank-1. The first and the second mode of the fUS tensor represents the width and depth of power-Doppler images (PDIs) are selected to be rank L_r . In the context of fUS, time information is often highly correlated. Thus, representing time information as rank-1 implies that changes over time are primarily captured using a single core tensor. This effectively reduces the dimensionality of the data while preserving the major temporal features. For the first and the second mode, width and depth information typically involves different spatial locations or regions. These pieces of information may exhibit more spatial variability, which will be useful to extract the expected areas from different components. More useful information will be retained using a low rank larger than 1.

Question VI-3: Compared to the CPD, what kind of differences do you observe in the extracted component(s) of interest?

Compared to the CPD, there are mainly two differences occurs after executing BTD on the fUS data. First, each component in BTD appears to correspond to multiple components. For instance, Figure 7 illustrates the combination of LGN and the visual cortex, while in CPD, as depicted in Figure 2,

component 1 represents LGN and component 15 represents the visual cortex individually. Almost all components in CPD results only correspond to one expected area. Another difference is that there will exist more unresolvable component in BTD such as components in Figure 8. These components do not correspond to any expected areas. This seldom occurs in CPD.

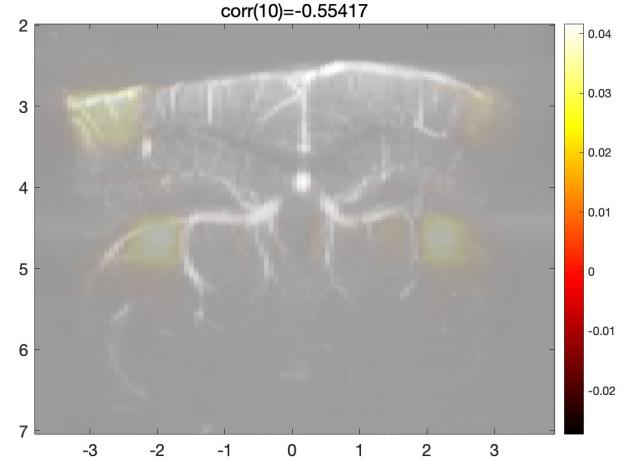


Fig. 7. One of the BTD components when $L = 2$

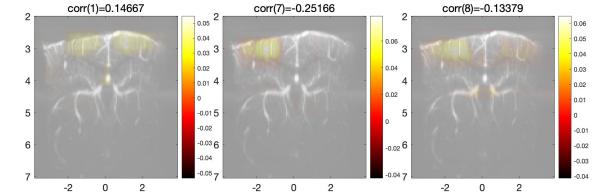


Fig. 8. unresolvable components in BTD when $L = 2$

Question VI-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.

Since each component in CPD has a rank of 1, while each component in BTD has a rank of L , the correspondence between BTD components and expected areas is inherently more complexed than that between CPD components and expected areas. For the unresolvable components in BTD, the reason is that BTD is a generalization of CPD, which can capture latent factors of rank higher than one in each component[1].

Theoretically, BTD offers increased complexity and provides more flexibility, essentially aggregating components of CPD. Therefore, achieving satisfactory outcomes with BTD might necessitate a higher number of experiments due to its stochastic nature of component aggregation.

Question VI-5: How did you determine the number of sources, and the factor matrices rank L ?

- Are you still able to extract a meaningful component when you select them differently?

- What happens when you run the BTD with different initializations and a fixed number of sources?

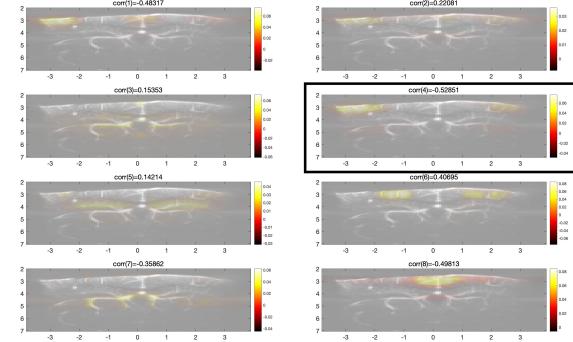


Fig. 9. Spatial maps of BTD when $R = 8$ and $L = 2$

• To determine the number of sources, we apply the same method mentioned in CPD algorithm. Noticably, in BTD algorithm, we should also consider the influence of the factor matrices rank L . We selected $L = 2$ and tested R values of 8, 12, and 20. Figure 9 shows the result for $R = 8$, indicating that certain components were not extracted properly, with only the visual cortex and blood vessels being successfully extracted. However, for R is larger than 15 (Figure 10), we observed repetitions of components and unexpected regions with relatively high correlation values. After synthesizing all the factors, we finally choose $R = 12$.

• To determine the factor matrices rank L , we initially examine the case where $L = 1$. In this scenario, the BTD algorithm aligns with the CPD algorithm. For each component after executing BTD, the first mode exhibits dimensions of $I_1 \times L$, and the second mode exhibits dimensions of $I_2 \times L$. When $L \geq 2$, the BTD algorithm offers enhanced flexibility and more complex spatial mappings compared to the CPD algorithm [3]. From Figure 5 and Figure 6 corresponding to $L = 2$ and $L = 3$ respectively, it can be noticed that it is much easier to extract the components points to the expected area when $L = 2$, in contrast to $L = 3$. Thus, it can be deduced that when L becomes larger, more information is incorporated including useless information. Furthermore, the performance of BTD is not sensitive to overestimation of L . BTD is more sensitive to the correct estimation of R (both overestimation and underestimation result in loss of accuracy)[1]. This means that correct choice of R is more crucial than the choice of L . Here, we choose $L = 2$ which presents good results.

• When employing different initializations with a fixed number of source, it can be predicted that the BTD results will exist obvious difference in comparison to previous outcomes. Figure 11 is an example of changing the initializations where $L = 2$. It is evident from the figure that components 9 and 12 correspond to the visual cortex area simultaneously, while no component corresponds to the LGN area. Despite the relatively high Pearson correlation coefficients between component 3 and component 6, these components do not align with the expected anatomical areas. Applying intializations in Figure 5 performs better results than Figure 11.

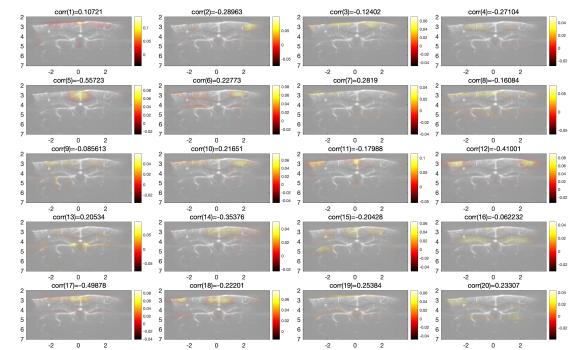


Fig. 10. Spatial maps of BTD when $R = 20$ and $L = 2$

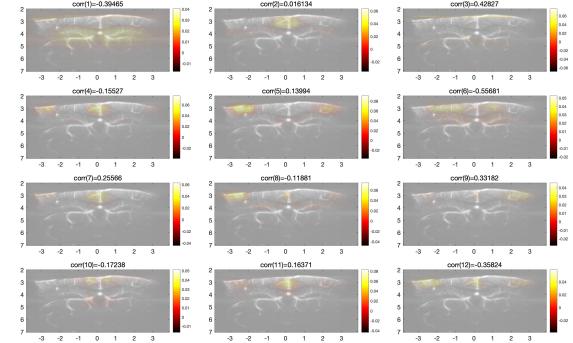


Fig. 11. Spatial maps of BTD when $L = 2$ with different initializations

VII. ADDITIONAL DISCUSSIONS

From the decomposition results by CPD and BTD, key information can be extracted using suitable initializations, a specified number of sources R , and factor matrices with rank L . Both algorithms are sensitive to the value of R , indicating that this parameter significantly impacts the results. Initializations also play a crucial role since reaching the global minimum is challenging in real-world data.

When applying BTD to 3-order fUS data, it can be observed that unexpected regions appear more frequently in comparison to CPD. This difference is attributed to the underlying principle of BTD, where its components have a rank of L_r rather than 1 for CPD. However, BTD offers greater flexibility in revealing the low rank features of the spatial mode than CPD. Consequently, a trade-off exists between the accuracy of spatial features and the presence of phantoms when making a choice between CPD and BTD.

A. Our Strategy for Enhancement

It has become evident that our utilization of BTD did not yield the much enhanced results we anticipated. This was primarily because once we increased the rank L_r to 2 for all components, the distinction between the LGN and visual cortex became obscured. Reflecting upon the BTD approach discussed in lectures, we recognize that we should have the flexibility to adjust the rank L_r of each BTD component independently. Ideally, we could assign a rank of 1 to the majority of components, while designating a higher rank to a select few. Such a configuration would allow us to discern the LGN and visual cortex at the rank-1 components, and observe more complex vascular structures elsewhere.

To this end, we utilized the work from TensorLab[4], specifically employing the ‘l1l’ function within it to solve the BTD- $(L_r, L_r, 1)$ problem, using an algorithm referred to as ‘minf’. Diverging from our iterative solution approach, TensorLab uses gradient based method. Specifically, upon formulating the optimization problem, it applies a linear backtracking search method to find the solution. Therefore, it tends to produce more stable results.

B. Results of additional experiment

In our initial experiments, an intriguing outcome has been observed. Specifically, when we escalate a L_r to a considerably high value, say 40, we obtain results as depicted in 12. Under these conditions, it appears that the decomposition with a rank of 40 closely approximates the correlation map we have derivedFigure 1 .

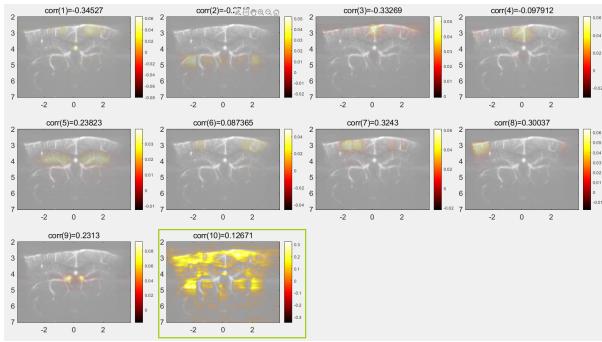


Fig. 12. Result of $L_r = 40$

Building on this foundation, our initial reaction was to reduce L_r , yet the outcomes did not align with our expectations.

Consequently, we contemplated adjusting two separate L_r values, as demonstrated in Figure 13. Within this configuration, we set $L_r = 1$ for r ranging from 1 to 8, $L_r = 10$ for $r = 9$, and $L_r = 5$ for $r = 10$. It was observed that in the component where $r = 10$, we successfully discerned the distribution areas of blood vessels, which exhibited a high correlation. Simultaneously, the other two regions, LGN and visual cortex, remained distinctly visible. This suggests that a targeted adjustment of ranks within a BTD can lead to the selective enhancement of specific features in the data while preserving the clarity of others.

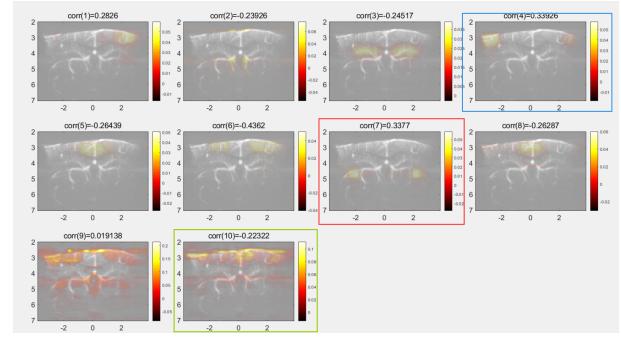


Fig. 13. Results of improved BTD

VIII. CONCLUSION

In this project, we investigated the use of tensor decomposition to extract features from functional ultrasound (fUS) imaging data. Specifically, with Canonical Polyadic Decomposition (CPD), we were indeed able to identify unique components corresponding to the LGN and visual cortex. However, the representation of blood vessels was dispersed across multiple components, and the approach was marred by instability, necessitating numerous experiments to achieve satisfactory results. When we employed Block Term Decomposition (BTD), we faced a challenge due to the increased complexity and degrees of freedom when the rank was elevated to 2; the LGN and visual cortex became difficult to delineate.

To address this, in our supplementary discussions, we explored the application of different ranks (L_r) to different components—mostly setting them to 1. Leveraging TensorLab’s more stable solving algorithm, we not only enhanced the repeatability of our results but also achieved excellent correspondence. In our experiments, we successfully isolated blood vessels using a component with a rank of 5.

REFERENCES

- Chatzichristos, C., Kofidis, E., Morante, M., & Theodoridis, S. (2019). Blind fmri source unmixing via higher-order tensor decompositions. *Journal of neuroscience methods*, 315, 17–47.
- De Lathauwer, L. (2008). Decompositions of a higher-order tensor in block terms—part ii: Definitions and uniqueness. *SIAM Journal on Matrix Analysis and Applications*, 30(3), 1033–1066.
- Hunyadi, B., Camps, D., Sorber, L., Paesschen, W. V., Vos, M. D., Huffel, S. V., & Lathauwer, L. D. (2014). Block term decomposition for modelling epileptic seizures. *EURASIP Journal on Advances in Signal Processing*, 2014(1), 1–19.
- Vervliet, N., Debals, O., & De Lathauwer, L. (2016). Tensorlab 3.0—numerical optimization strategies for large-scale constrained and coupled matrix/tensor factorization. In *2016 50th asilomar conference on signals, systems and computers* (pp. 1733–1738). <https://www.tensorlab.net/>.