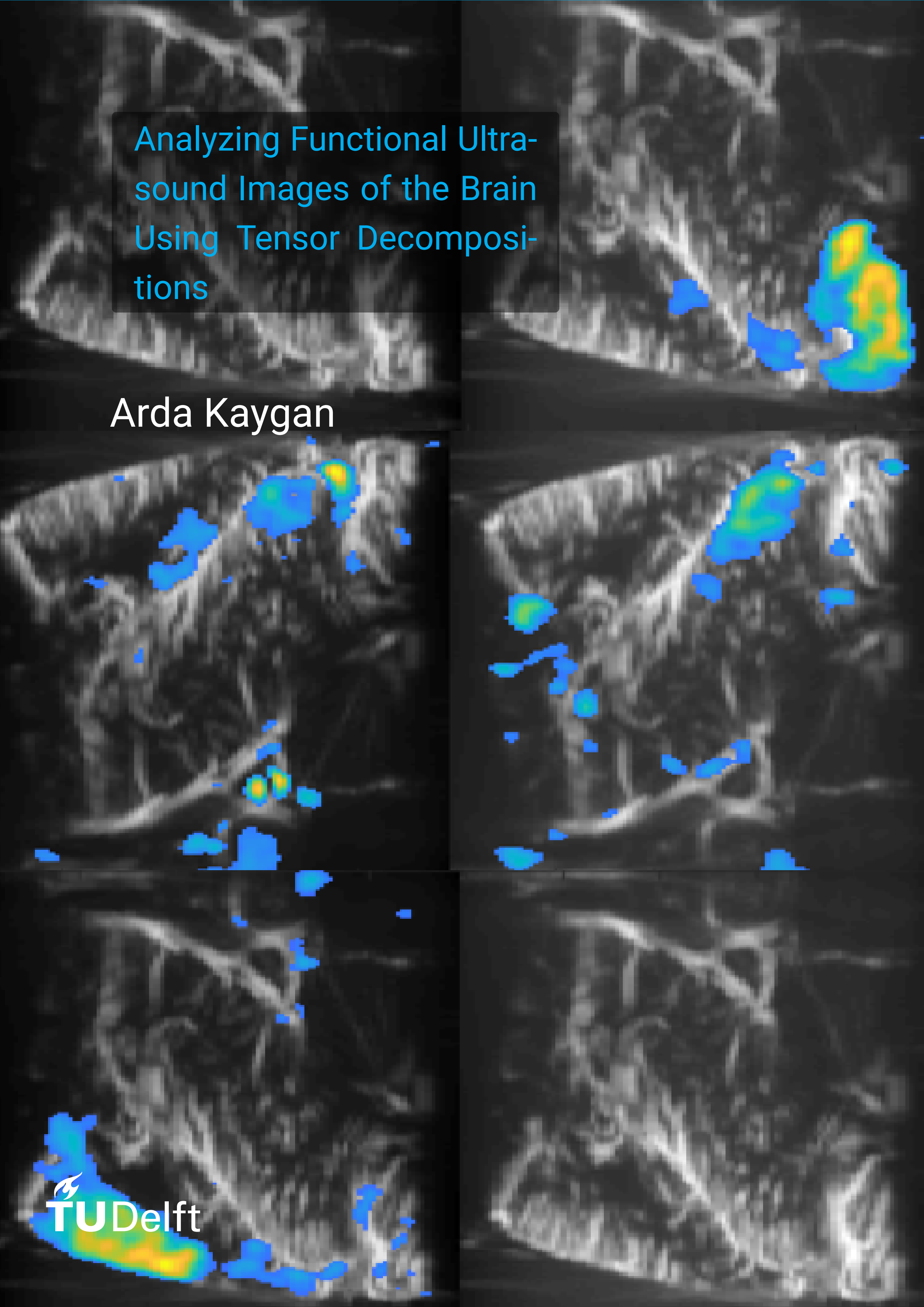


Analyzing Functional Ultrasound Images of the Brain Using Tensor Decompositions

Arda Kaygan

Analyzing Functional Ultrasound Images of the Brain Using Tensor Decompositions

A thesis by

Arda Kaygan

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday December 3$^{\text{rd}}$, 2021 at 3:00 PM.

Student number:	4812026	
Thesis committee:	Prof. dr. ir. A.J. v.d. Veen,	TU Delft, supervisor
	Dr. B. Hunyadi,	TU Delft, supervisor
	Dr. P. Kruizinga,	Erasmus MC, external research expert
	A. Erol,	TU Delft, supervisor

TU Delft
Delft
University of
Technology

Acknowlegments

First of all I would like to thank to my supervisor Bori Hunyadi, who guided me troughout this journey with her never ending support. I would also like to thank to Aybüke Erol who joined her in guiding me from the very beginning. I feel exceptionally fortunate to be surrounded by such fruitful guidance, allowing me to bring together this work despite all the challenges. I would also like to express my gratitude to Alle Jan van der Veen for his guidance especially during the first months, introducing me to the CAS group to execute my project.

Furthermore I would like to thank to people of CUBE for their crucial support, letting me be a part of their incredible work. In particular I would like to thank to Pieter Kruizinga who kept providing inspiring comments during our meetings, guiding me in working with fUS as the substance of my project.

Last but not least I would like to thank to my parents whose love and support mean everything to me. They have been together with me in every challenge I tackled in life, and this was no exception.

Abstract

Functional ultrasound (fUS) is a neuroimaging modality that offers high spatial and temporal resolution while also providing portability. In this thesis, neuroimaging data acquired with fUS at Center for Ultrasound and Brain imaging at Erasmus MC (CUBE) is processed. Due to the fact that fUS data is inherently multidimensional, we propose using tensor decompositions, tensors here referring to generalizations of matrices, for processing of fUS data.

We define two main research questions regarding fUS data analysis. First, for compressing the large-scale raw beamformed fUS data, we apply sequentially truncated multilinear singular value decomposition. This compression method is compared against ensemble averaging used in the conventional pipeline, and shown to provide a higher compression rate while preserving more temporal resolution for specific ranks. Furthermore, it is observed to denoise the data, resulting in a more precise extraction of the active region of Superior Colliculus using correlation maps.

Secondly, in order to investigate the advantage of multi-slice processing that incorporates 3-D information, blind-source separation methods are applied to single slice and two-slice fUS recordings. After applying independent component analysis (ICA) to the matricized data as a benchmark method, block term decomposition (BTD) is used as a way of processing the data as it is, in its natural 3-D structure without vectorization. Through a simulation study, it is shown that the method is able to separate two images even when using a rank that is lower than the true rank, as well as in noisy conditions. Subsequently, BTD is applied to real 4-D fUS data formed by concatenation of slices in a new dimension. However, this method is seen to perform worse than single slice ICA in terms of extracting the active regions. In order to amplify common information between slices, a new 3-D data structure is then formed by summing the fUS data of two slices. For extraction of this common information, a BTD is then applied to the aggregate 3-D data. The findings of this decomposition reveal that both taking a longer portion of single slice data and incorporating the second slice helps to achieve better results.

Contents

Acknowlegments iii

Abstract v

1 Introduction 1
- 1.1 Motivation: Why Tensors? 1
- 1.2 Problem Statement 2
- 1.3 Outline 2
- 1.4 Basics of Tensors 2

2 Data 5
- 2.1 Functional Ultrasound 5
- 2.2 Image Acquisition and Processing Pipeline 6
- 2.3 Experimental Paradigm 6
- 2.4 Dataset 7

3 Compression via Sequentially Truncated Multi-linear Singular Value Decomposition 9
- 3.1 Motivation 9
- 3.2 Signal Model 10
- 3.3 Theory 10
- 3.4 Results 11
  - 3.4.1 The Stimulus Signal 11
  - 3.4.2 Preservation of Correlated Region using ST-MLSVD Compression 12
  - 3.4.3 Preservation of Temporal Average Response using MLSVD Compression 14
- 3.5 Discussion 19

4 Blind Source Separation in Multi-Slice Functional Ultrasound Data 21
- 4.1 Motivation 21
- 4.2 Signal Model Revisited 21
- 4.3 Independent Component Analysis: Theory and Expectations 22
- 4.4 ICA Results 22
  - 4.4.1 The Stimulus Signal 22
  - 4.4.2 Pre-processing 23
  - 4.4.3 Single Slice 23
  - 4.4.4 Multi-Slice 26
  - 4.4.5 Limitations 27
- 4.5 Signal Model Revisited Again 28
- 4.6 Block Term Decomposition: Theory and Expectations 29
- 4.7 BTD Results 29
  - 4.7.1 Simulation 29
  - 4.7.2 4-D Decomposition 34
  - 4.7.3 Mixture Decomposition 35
- 4.8 Discussion 41

5 Conclusion 43
- 5.1 Future Work 44

1

Introduction

Extraction of information regarding the functioning of the brain is believed to have an immense value for a wide range of applications. These include diagnostics and treatment of neurological diseases and disorders as well as realization of brain-computer interfaces.

Functional ultrasound (fUS) imaging is an emerging method of functional imaging that brings high spatial and temporal resolution together in a way that was not possible for its predecessors to provide. Unlike fMRI, which also provides high spatial resolution, fUS imaging allows for lightweight probes to be used, making it highly mobile. As a functional neuro-imaging modality, fUS captures the spatiotemporal changes in Cerebral Blood Volume (CBV) in response to the changes in brain activity through a process known as neurovascular coupling. As such, the inherent structure of data produced by fUS is multi-dimensional which can be captured by tensor representations.

Tensor-based signal processing constitutes a set of methods for processing multi-dimensional data that gained popularity in recent years, including applications in biomedical signal processing due to the inherently multidimensional structure of medical data [1] [2]. Tensors here refer to higher order generalizations of matrices, allowing for an alternative representation of data [3].

In this thesis, two uses of tensor-based signal processing will be explored. First, tensor decompositions will be utilized for compressing raw beamformed fUS images. Second, tensor decompositions will be utilized for applying blind source separation algorithms that process data as in its multidimensional structure.

1.1. Motivation: Why Tensors?

The motivation for using tensors to represent information produced by brain stems from the fact that this information spans multiple dimensions including 3-dimensional space, time as well as others including frequency, trials, segments or subjects. Indeed, brain activity that can be captured in a functional neuro-imaging experiment is inherently spread across time and space, and can be further segmented or processed to be represented in additional dimensions, named as modes in the literature [4]. Capturing variations in these different modes as well as their interactions with each other is possible through tensor representations, which gives us the following crucial advantages;

- Low-rank structuring uncovered by tensor decompositions that reveal the redundancies in data that were previously not apparent. Compression can be achieved by eliminating these redundancies, utilizing the tensor form [5] [6].

- Extraction of information via finding a unique solution to a decomposition is easier for tensors as such solutions exist in milder conditions when data is represented in tensor form. [7].

- Extraction of information via finding a global solution in noisy conditions is easier for tensors as the multi-dimensional structure preserved by tensor representations impose additional constraints, helping the decomposition algorithm to navigate through local minima generated by noise [8].

These three advantages form the basis of the motivation behind this thesis. The validity of the first advantage is investigated in Chapter 3 and the validity of the second and the third is investigated in Chapter 4.

1.2. Problem Statement

The ultra-fast acquisition sequence produces great amount of fUS data to analyse. The first question that will be addressed in this work is thus;

RQ 1: Is there a way to reduce the data size considerably while preserving the signal related to the functional response of the brain as well as high frequency information?

In order to address this problem, Sequentially Truncated-Multilinear Singular Value Decomposition (ST-MLSVD) as a higher order singular value decomposition will be used on data consisting of raw beamformed frames.

In capturing data used in this work, imaging is done with probes that capture 2-D frames on multiple slices. Thus, it is desirable for an analysis method to incorporate 3-D information and utilize the true spatial structure of the brain. Since what we have is only data captured in 2-D for different slices, the second question to be addressed is;

RQ 2: Compared to processing slices separately, can simultaneous processing of different slices be advantageous, by allowing us to extract new information or allowing us to extract previously available information more easily?

In order to address this question, Block Term Decomposition (BTD) will be applied to the data as a blind source separation algorithm to extract information about the active regions in the brain. The results will then be compared to those obtained from processing 2-D slices separately, using Independent Component Analysis (ICA) as an established method for blind source separation on vectorized data [9] [10].

1.3. Outline

In order to get a better understanding of the problems at hand, the data acquired during the experiment will be described in Chapter 2. In this chapter, the process of ultra-fast acquisition will be explained along with the biological base of what is being measured by fUS will be established. The processing pipeline to acquire Power Doppler Images (PDIs) will be described.

In chapter 3 the first question related to efficient compression of fUS data will be addressed. First, the motivation behind compression of raw beamformed images will be described. Starting with the signal model, the usage of low-rank tensor decompositions to compress raw beamformed fUS data will be justified. The research question presented will then be answered by illustrating that the functional response and high frequency features of the data is preserved while achieving further compression compared to the conventional method of ensemble averaging for chosen ranks that are found to be feasible for the problem at hand.

In chapter 4, the second question related to multi-slice processing of fUS data will be addressed. First, the motivation behind incorporating multi-slice information will be described. The signal model given in Chapter 3 will be revisited, pointing out to different sources of brain activity present at the same time. In order to separate these sources, blind source separation algorithms will be used. Single-Slice ICA applied on two slices separately will be given as a benchmark method. In order to overcome the limitation of not including 3-D information by processing slices separately, multi-slice ICA will then be introduced by concatenating pixels of each slice in space before decomposition. However in both of these approaches, vectorization will still be used leading to loss of information related to spatial structure in images. To overcome this, Block-Term Decomposition (BTD) will be introduced as a way of processing the data as it is. It will be first applied in a simulation study, illustrating blind source separation capability of the approach on the task of separating two different photographs. Then, BTD will be applied to 4-D data formed by stacking slices on a new dimension. Lastly, BTD will be applied to 3-D data formed by summation of two slices, and results will be evaluated using a binary mask of the region Superior Colliculus (SC) obtained from ICA trials.

1.4. Basics of Tensors

Tensors are generalizations of matrices and vectors, which can be regarded as second order and first order tensors respectively. Indeed, by defining an outer product between N vectors, an Nth-order tensor can be described, which is characterized by one of these vectors at each mode. This operation can be denoted as;

$$\boldsymbol{X} = u^{(1)} \times u^{(2)} \times \ldots \times u^{(N)} \tag{1.1}$$

where $\boldsymbol{X} \in R^{I_1 \times I_2 \times \ldots \times I_N}$ with $u^{(n)} \in R^{I_n}$ corresponding to a vector in mode n.

The tensor $\boldsymbol{X}$ defined here is a rank-1 tensor, formed by the outer product between a single group of vectors. This definition can be extended to higher ranks via a linear combination of such vector outer-products, as shown below:

$$\boldsymbol{X} = \sum_{r=1}^{R} u^{(1)r} \times u^{(2)r} \times \ldots \times u^{(N)r} \tag{1.2}$$

Equation 1.2 is called as Canonical Polyadic Decomposition (CPD), and the minimum number of R that returns the original tensor $\boldsymbol{X}$ is defined as the tensor rank [6].

The n-mode product can be regarded as a generalization of the outer product, defined as a product between an Nth order tensor and a matrix. This operation is denoted as;

$$\boldsymbol{Z} = \boldsymbol{X} \times_n A \tag{1.3}$$

where $X \in R^{I_1 \times I_2 \times \ldots \times I_N}$, $A \in R^{J \times I_n}$ and $Z \in R^{I_1 \times I_2 \times \ldots \times I_{n-1} \times J \times I_{n+1} \times \ldots \times I_N}$. Elements of $\boldsymbol{Z}$ can be computed as;

$$z_{i_1 \ldots i_{n-1} j i_{n+1} \ldots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 i_2 \ldots i_N} a_{j i_n} \tag{1.4}$$

These two products will form the base of the decompositions that will be included in this thesis, which are Sequentially Truncated Multilinear Singular Value Decomposition (ST-MLSVD), Independent Component Analysis (ICA) and Block Term Decomposition (BTD) in their respective order. In ICA, sum of outer products of vectors that form the spatial and temporal signatures of each component are found. In ST-MLSVD and BTD, n-mode product is used for defining the interaction between signatures related to different modes such as time and multiple spatial dimensions by taking n-mode product between a core tensor and factor matrices. These factor matrices are those that include signatures as columns which form information in these modes such as activation pattern of a spatial map along time included in a factor matrix related to time mode.

2

Data

2.1. Functional Ultrasound

Functional ultrasound captures changes in Cerebral Blood Volume (CBV), which is defined as the volume of blood within a given quantity of brain tissue. In ultrasound, this quantity is reflected by the number of red blood cells that lie within a region as sources of contrast [11]. The main principle used in neuro-imaging is that brain cells need more oxygen in the case of increased activity, thus attracting more oxygenated blood from the vessels. Thus CBV increases in regions that become more active, allowing us to see where the activity is located in fUS scans. The coupling between neural activity and blood flow is called hemodynamic coupling, which is often utilized neuro-imaging [12].

As indicated in Section 1, fUS provides high temporal and spatial resolution by utilizing an ultra-fast acquisition scheme while also providing portability unlike other mediums of imaging. Figure 2.1 shows where fUS stands in terms of these properties compared to other imaging methods.

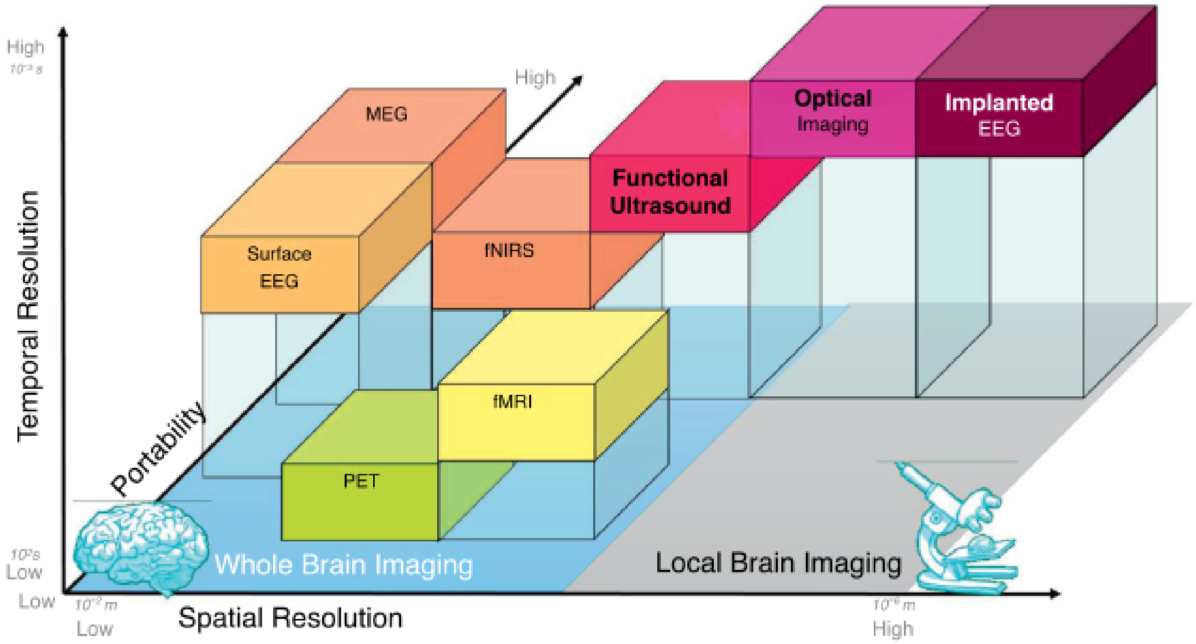


Figure 2.1. Comparison of fUS with other imaging modalities regarding spatial resolution, temporal resolution and portability.

The privileges of fUS allows it to be regarded as a promising imaging technique for achieving less-invasive and portable brain-machine interfaces [13]. In [13], data acquired by fUS is shown to be used for prediction of movement response decoding the brain, proving itself to be a key candidate for development of such technologies.

2.2. Image Acquisition and Processing Pipeline

In the traditional ultrasound acquisition sequence, imaging is achieved using focused waves whose backscattered echo signals are used to form a single line of the image. The acquisition sequence used in this work uses plane waves instead, greatly increasing frame rate by only requiring echo signal from one emission of a plane wave to construct a whole image [11].

The conventional processing pipeline of acquired raw beamformed images are as follows. Acquired images are first subjected to clutter filtering, as both slow-moving tissue-clutter signal and fast-moving blood signal is captured in these images. In order to extract the blood signal which is valuable for our purposes, first a high pass filter in time with a cut-off frequency of 60 Hz is applied. Then, a spatiotemporal SVD filter that exploits the fact that blood and tissue-clutter signals are included in different subspaces is applied, thus forming images that largely include information related to blood movement [14].

The images are then absolute squared, forming Power Doppler Images (PDIs). In the conventional pipeline, images are then averaged over each ensemble, and the word PDI in the literature is often used to refer to the result of this averaging. In this work, we will make the distinction clear in the following chapters as the ensemble averaging step will be replaced by a low-rank multilinear decomposition in Chapter 3.

2.3. Experimental Paradigm

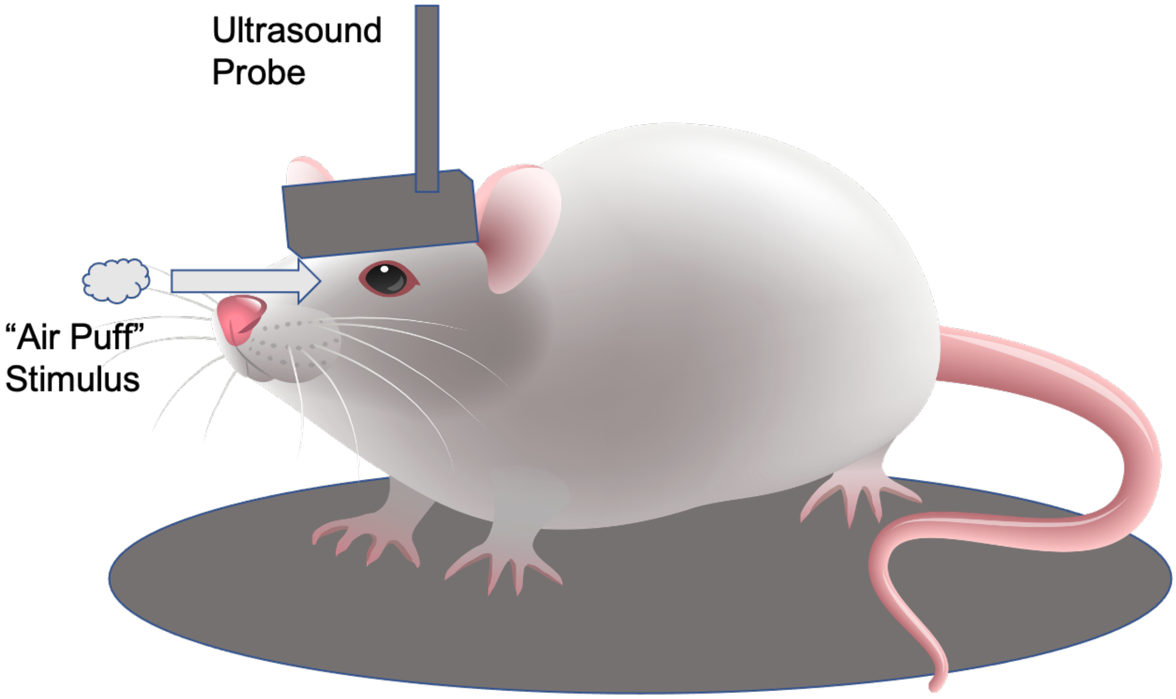


Figure 2.2. Experimental Paradigm: Air Puff’s of duration 30 ms and 15 ms are applied to the left eye of the mouse subject.

The experimental paradigm used here includes successive air puffs of length 15 ms and 30 ms applied to the left eye of the mouse during a 506 second recording. It is known that a 30 ms air puff is expected to generate more pressure, making it a more intense stimulus compared to 15 ms. Neither 30 ms or 15 ms air puffs are expected to be creating discomfort for the mouse. Rather, they’re expected to attract the attention of the mouse, triggering a response in its brain related to direction of attention.

Air puffs are regarded as tactile stimuli, received as somatosensory information by the brain [15]. Superior Colliculus (SC) is a region known for its integration of visual and somatosensory information before evoking motor responses [16]. It is also known for its role in direction of attention [17]. It is also known to relay information to other parts of the brain such as Thalamus [18].

In this experiment, the functionalities of SC described above is expected to be observed. That is, its activation by the air-puff stimulus, leading to an activation in motor cortex as the mouse responds to the stimuli with saccade or whisker movements, orienting its attention. An activation in Thalamus is also expected to be observed, taking place after activation in SC.

2.4. Dataset

Data used in this work consists of 398728 frames captured during a 506 second long recording per slice. These frames are acquired in the form of ensembles, each of which consists of 197 frames captured with a sampling frequency of 800 Hz. Between acquisition of each ensemble, a time gap of 3.75 milliseconds exists, making the ensemble capture rate (PDI sampling rate) 4 Hz.

Recordings are repeated with the exact same experimental paradigm being used. In total, 13 slices on the sagittal plane are recorded, 3 of which will be used in this work. On these slices, the region SC is clearly evident which forms the motivation to use these particular slices.

Original dimensions of data is $N_z^* = 256, N_x^* = 128, N_y = 13, N_F^* = 2024 \times 197 = 398728$. Images are cropped before processing, thus the dimensions of data processed in this work are $N_z^* = 168, N_x^* = 98, N_y = 3, N_F^* = 398728$. Cropped mean power Doppler image of 3 slices used in this work is shown in Figure 2.3/

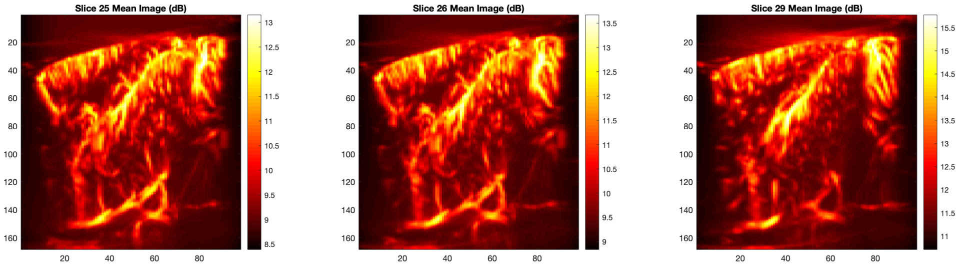


Figure 2.3. Mean images for Slice 25, 26 and 29.

The first slice on the left denoted as Slice 25 is positioned at 1.1 millimeters away from bregma, while the second slice in the middle denoted as Slice 26 is positioned at 1 millimeter away from bregma, with 100 microns distance to the adjacent Slice 25. Lastly, third slice on the right denoted as Slice 29 is positioned at 0.7 millimeters away from bregma [19].

3

Compression via Sequentially Truncated Multi-linear Singular Value Decomposition

3.1. Motivation

The frame rate of ultra-fast Doppler acquisition allows for capturing functionality of the brain at a resolution of up to 1.25 milliseconds. However, due to requirements in storage and computational complexity, a tiny fraction of this temporal resolution is utilized by taking the average of every ensemble of frames. The typical ensemble rate is about 200 frames, thus such averaging reduces the frame rate to as low as 4 Hz. This averaging step is also placed in conventional pipeline to increase signal-to-noise ratio (SNR), however it is unclear if such large window is needed for this purpose.

One fundamental tool that can be used to reduce size of data to be stored is Singular Value Decomposition (SVD), in which each image is vectorized into single dimensional vectors to form a data matrix with columns corresponding to each frame. SVD can be then used to decompose the data into three matrices. By truncating the resulting factor matrices, compression can be achieved by storing only a part of values that represent the data [20] [21] [22] [23].

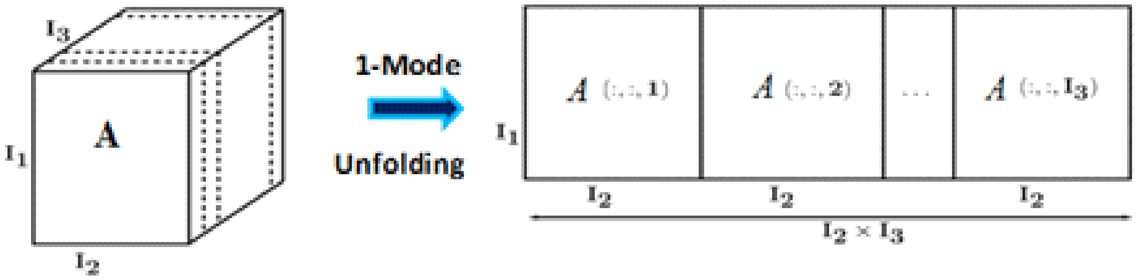


Figure 3.1. Illustration of unfolding in the first mode. In the case of unfolding for SVD in neuroimages, I_2 and I_3 correspond to spatial dimensions forming each image, whereas I_1 corresponds to the number of frames.

However, unfolding the data with such vectorization prevents such decomposition from exploiting the low rank structure in space, leading to loss of information [24]. For instance, a local active region formed by neighbouring pixels appearing on an image can be represented by multiplication of sparse matrices, which can be truncated to achieve compression as the region can be represented by outer product of few vectors. If we vectorize the image, the pixels that are neighbouring each other end up on locations further away from each other, preventing such possibility to utilize structuredness in space [25].

Another possiblity could be to consider each image as a matrix, and apply SVD compression separately to each frame. However in this case, information related to time dimension is not incorporated, preventing us from utilizing any redundancies related to this dimension.

In this work, a higher order singular value decomposition is proposed instead of ensemble averaging, in order to reduce the storage requirement of data captured at ultra-fast frame rate. By using a higher order decomposition without unfolding, low-rank structure in each of the dimensions are preserved allowing us to exploit redundancies that appear from interaction of all these dimensions. Since the frame rate is not sliced as in the case of ensemble averaging, higher frequency features are preserved while leading to more compression.

A critical constraint in this compression scheme is the requirement for preserving parts of the signal related to subjects' response to the stimulus, appearing as correlated time-series observed in specific regions of the brain. This constraint poses a challenge in two aspects. First, it requires a definition and evaluation metric for the preserved response. Second, it prompts for consideration of cases where main image features such as anatomical features are preserved, i.e. the algorithm works efficiently as a generic image compression algorithm, whereas parts of the signal correlated with stimulus time course is lost, i.e. functional response of the brain to the stimulus is not preserved.

3.2. Signal Model

Below, a simple signal model is presented;

$$\boldsymbol{X} = \boldsymbol{S} + \boldsymbol{N} \tag{3.1}$$

Let $\boldsymbol{X}$ be measured data represented in $R^{N_z \text{ x } N_x \text{ x } N_F}$. $\boldsymbol{S}$ here represents the signal related to functional brain activity which includes information that is desired to be extracted whereas $\boldsymbol{N}$ corresponds to noise.

The main assumption in this model is that S is characterized by a local neighbourhood of pixels acting in a correlated manner in time and thus is structured in time and space. Therefore it can be represented by a low-rank multilinear model unlike $\boldsymbol{N}$, which shows high rank characteristics, distributed across the data as uncorrelated noise [26]. Thus it is believed that $\boldsymbol{S}$ can be extracted and preserved by a low-rank multilinear decomposition such as a higher order SVD.

3.3. Theory

After extraction of the mean image, each recording of a single slice can be represented with the 3-D tensor $\boldsymbol{X} \in R^{N_z \text{ x } N_x \text{ x } N_F}$. If the time domain is segmented, the segments could be stacked in a fourth dimension, leading to a 4-D tensor of $\overline{\boldsymbol{X}} \in R^{N_z \text{ x } N_x \text{ x } N_f \text{ x } N_s}$ where $N_f N_s = N_F$. In this framework, a segment tensor is denoted as $\overline{\boldsymbol{X}}_s \in R^{N_z \text{ x } N_x \text{ x } N_f}$ where $s = 1, ..., N_s$.

To compress, each segment can be decomposed into a feature core tensor and factor matrices learned from all of the data;

$$\overline{\boldsymbol{X}}_s = \overline{\boldsymbol{G}}_s \times_1 U_z \times_2 U_x \times_3 U_f \tag{3.2}$$

Where $\overline{\boldsymbol{G}}_s \in R^{I_z \text{ x } I_x \text{ x } I_f}$ form the features that lay on the space spanned by base matrices $U_z \in R^{N_z \text{ x } I_z}$,$U_x \in R^{N_x \text{ x } I_x}$ and $U_f \in R^{N_f \text{ x } I_f}$. By concatenating core segment tensors on the fourth dimension, $\overline{\boldsymbol{G}} \in R^{I_z \text{ x } I_x \text{ x } I_f \text{ x } N_s}$ is formed. This tensor is shown to be equivalent to a projection from all data in [27] such that;

$$\overline{\boldsymbol{X}} = \overline{\boldsymbol{G}} \times_1 U_z \times_2 U_x \times_3 U_f \tag{3.3}$$

The main idea behind compression using this framework is to be able to store the core tensor and factor matrices. The compression achieved with such storage is given by the fraction below. The term on the rightmost side is accounted for storing the mean image;

$$\frac{I_z I_x I_f N_s + N_z I_z + N_x I_x + N_f I_f + N_z N_x}{N_z N_x N_f N_s} \tag{3.4}$$

Sequentially truncated multilinear singular value decomposition (ST-MLSVD) described in [28] provides a higher order generalization for truncated SVD using a sequential truncation scheme. In this approach, approximations of factor matrices U's that approximately satisfy the minimization problem in Equation 2.4 is found. Thus, a close reconstruction after projection and back-projection is aimed at;

$$\min_{U_z,U_x,U_f} \quad ||\overline{\boldsymbol{X}} - (\overline{\boldsymbol{X}}) \times_1 (U_z^T U_z) \times_2 (U_x^T U_x) \times_3 (U_f^T U_f)||^2_{Frob} \tag{3.5}$$

One method used for approximating these factor matrices is based on truncated SVD's of unfoldings of $\overline{\boldsymbol{X}}$. However, this truncation method is found to have a high upper error bound when at least one of the projectors do not provide a good approximation of the actual non truncated orthogonal projection, such as in the cases where one mode is heavily truncated [28].

In [28] another method for obtaining truncated factor matrices is found by assuming the projections in Equation 2.4 are ordered in a particular sequence of modes. Letting this order to be $p = N_z, N_x, N_f$, the minimization problem can be written as;

$$\begin{aligned} \min_{U_z,U_x,U_f} \quad & ||\overline{\boldsymbol{X}} \times_1 (1 - U_z^T U_z)||^2_{Frob} + ||\overline{\boldsymbol{X}} \times_1 (U_z^T U_z) \times_2 (1 - U_x^T U_x)||^2_{Frob} \\ & + ||\overline{\boldsymbol{X}} \times_1 (U_z^T U_z) \times_2 (U_x^T U_x) \times_3 (1 - U_f^T U_f)||^2_{Frob} \end{aligned} \tag{3.6}$$

One may notice that from left to right, terms inside the summation are likely to take up lesser part of the minimization, as they involve more bases that are subject to this optimization problem. The main idea behind approximation of the optimal solution in ST-MLSVD is to optimize these terms separately in sequence that is formed here, starting with finding U_z which is defined at the first term in this particular order. Then the error for projection on the second mode defined by factor matrix U_x is minimized using the projected data using U_z. Lastly, error of U_f is minimized on the projected data using the two bases came before. Thus, the recursion follows as in algorithm given in 3.2;

```
$\hat{\mathcal{S}} \leftarrow \mathcal{A}$
for $k = p_1, p_2, \ldots, p_d$ do
    { Compute the left singular vectors of $\hat{\mathcal{S}}_{(k)}$ }
    $\hat{\mathcal{S}}_{(k)} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$, with $U_1 \in \mathbb{R}^{n_k \times r_k}$
    $\hat{U}_k \leftarrow U_1$
    $\hat{\mathcal{S}}_{(k)} \leftarrow \Sigma_1 V_1^T$
end for
```

Figure 3.2. ST-MLSVD Algorithm as described in [28]. Curly A stands for the input, $\overline{\boldsymbol{X}}$.

It is apparent that the algorithm and the error it gives is dependent on the chosen order of modes. ST-MLSVD method is claimed to generate less error in most cases compared to methods involving parallel truncation for the same ranks. When one of the modes are truncated to 1, it is guaranteed to give a better approximation compared to the case of parallel truncation with the same ranks.

3.4. Results

3.4.1. The Stimulus Signal

In order to evaluate preservation of valuable information in fUS data, preservation in both space and time should be considered. Thus, information on where and when there is a functional response must be conserved. One method to extract this information is by looking at how each pixel's time course is correlated using Pearson correlation with the time course of stimuli given to the mouse. Such time course can be modeled by a signal at the time scale of raw beamformed frames with a function including delayed rectangular pulses of length 15 ms and 30 ms representing air puffs given to the mouse. However, a more biologically plausible signal including estimated responses to each stimulus can be used to assess correlation as well, which is the route chosen in this work. Here, a hemodynamic response function (HRF) estimated from fMRI data peaking at 100ms shown in Figure 3.3 is convolved with the signal including delayed rectangular functions with each corresponding to a stimulus. In the end, the stimulus signal given in Figure 3.4 is achieved, providing us an approximation of brain's response to the experimental paradigm used. It should be noted that the HRF used here is a version derived from canonical estimate, and the real impulse response to the stimulus can be varied depending on the subject and the region. The parameters of this HRF are deliberately set to give a quick rising peak as ideally quick response of the brain only visible in raw beamformed time scale is aimed to be extracted.

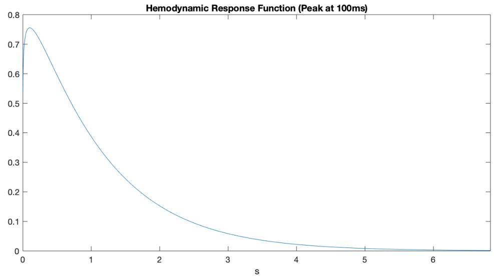


Figure 3.3. Hemodynamic Response Function (HRF), peaking at 100ms used in this chapter.

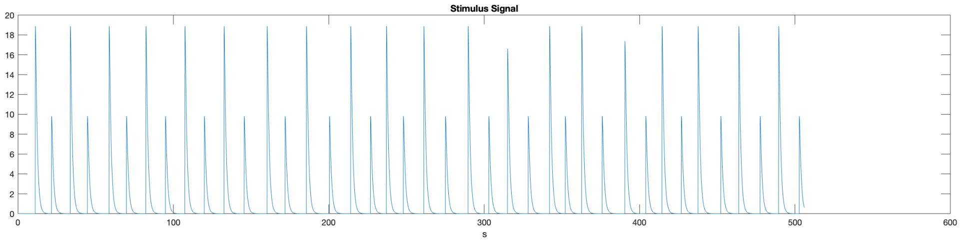


Figure 3.4. Stimulus signal computed by convolution of HRF with the stimulus time course with delayed rectangular functions that correspond to each 30 ms or 15 ms air puff.

As a result of convolving with longer rectangular pulses, stimulus signal shown in 3.4 involves higher intensity peaks for 30ms air puffs compared to 15 ms peaks. This is believed to be physically and biologically plausible, as the 30 ms air puff is known to generate more air pressure sensed by the mouse, giving rise to a more intense response.

In Subsection 3.4.1, correlation maps indicating Pearson correlation coefficient for each pixel with signal given in Figure 3.4 is used to evaluate preservation of the functional response in spatial domain. In order to evaluate preservation of temporal properties other than correlations, average response signals calculated by placing an averaging window at the onset and afterward of each stimulus is used. In Subsection 3.4.2, these average response signals acquired from reconstructed data after MLSVD and PDI's are compared with the average response signal captured from the original data. Normalized Mean Square Error (NMSE) as a measure of similarity is used to asses whether temporal features of the response is preserved.

3.4.2. Preservation of Correlated Region using ST-MLSVD Compression

Using Pearson correlation with the delayed stimulus signal of each pixel, regions that has a time course correlated with the stimuli can be observed in original data;

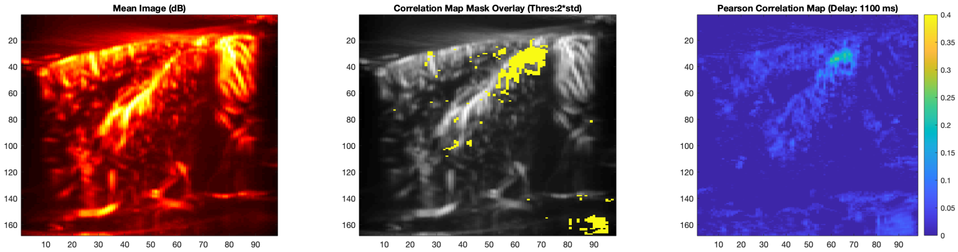


Figure 3.5. Overview of original data to be preserved in 3 plots. Left) Mean image. Middle) Thresholded correlation map overlaid on mean image. Correlation map is first computed by computing every pixel's time course's Pearson correlation with the stimulus signal. The obtained map is then thresholded using 2 times standard deviation as a threshold before being overlaid on the mean anatomical image. Right) Correlation map showing Pearson correlation values for each pixel and the stimulus signal.

Figure 3.5 illustrates spatial and temporal information that is aimed to be preserved in 3 plots. First, the mean image shows the main anatomical features of the data including vessels. On the second image the relatively high correlated region of SC is overlaid on the mean image as a thresholded mask, indicating the location of the active region. Third, total correlation map indicating the level of correlation for each pixel is shown.

In this work, both anatomical features and information regarding correlated regions in time and space is aimed to be preserved via MLSVD.

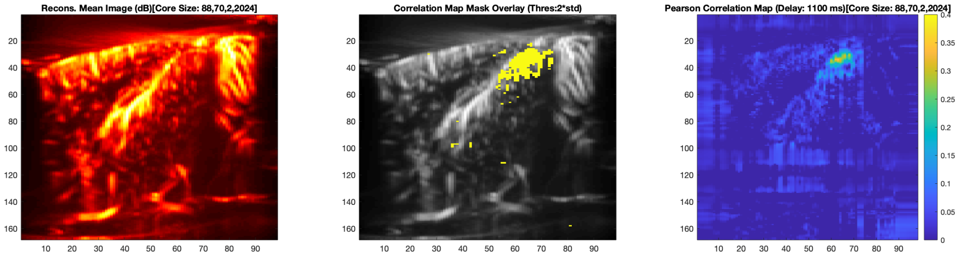


Figure 3.6. Overview of data reconstructed after ST-MLSVD with ranks 88,70,2,2024. Left) Mean image. Middle) Thresholded correlation map overlaid on mean image. Correlation map is first computed by computing every pixel's time course's Pearson correlation with the stimulus signal. The obtained map is then thresholded using 2 times standard deviation as a threshold before being overlaid on the mean anatomical image. Right) Correlation map showing Pearson correlation values for each pixel and the stimulus signal.

Figure 3.6 shows the result of truncation to ranks $I_z = 88, I_x = 70, I_f = 2$ from $168, 98, 197$ in first three modes, shrinking the size of the data to 0.38% of its original size. The data is then reconstructed using the core tensor and factor matrices. The anatomical features of the mean image is shown to be preserved since the mean was extracted before decomposition and added back after reconstruction. Active region of SC is clearly visible in correlation map, denoting preservation of neural response to the stimulus. Furthermore, a region that appears on bottom right corner of the original overlay which we know to be outside of the brain seems to have disappeared indicating the denoising capability of ST-MLSVD. Pearson correlation values are also seen to be increased for active pixels. This is due to the low-pass filtering effect of truncation in time, leading to increased correlation with convolved stimulus signal. Despite the preservation of correlated region shape, line-shaped artifacts in horizontal and vertical directions are visible. This is due to the fact that MLSVD is based on SVD truncations in each mode. Each SVD is done on matrix generated by flattening the tensor in the particular mode, taking fibers along the dimension of flattening as columns of data. When data is tensorized again, artifacts due to missing singular vectors of each fiber is reflected in these lines along the truncated mode.

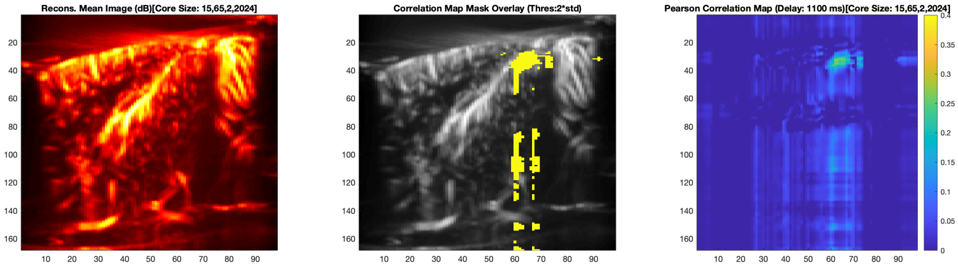


Figure 3.7. Overview of data reconstructed after ST-MLSVD with ranks 15,65,2,2024. Left) Mean image. Middle) Thresholded correlation map overlaid on mean image. Correlation map is first computed by computing every pixel's time course's Pearson correlation with the stimulus signal. The obtained map is then thresholded using 2 times standard deviation as a threshold before being overlaid on the mean anatomical image. Right) Correlation map showing Pearson correlation values for each pixel and the stimulus signal.

Figure 3.7 shows truncation to ranks $I_z = 15, I_x = 65, I_f = 2$ preserving the shape of active region SC. However, correlation map shows more severe artifacts in this case compared to Figure 3.6, with the shape of highly correlated regions being affected from the line-like artifacts and false positive highly correlated regions are generated.

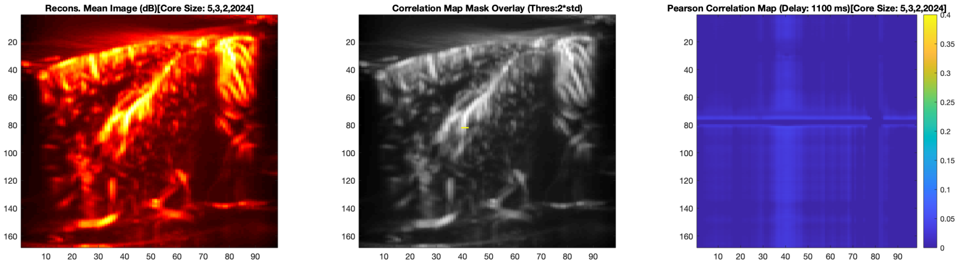


Figure 3.8. Overview of data reconstructed after ST-MLSVD with ranks 5,3,2,2024. Left) Mean image. Middle) Thresholded correlation map overlaid on mean image. Correlation map is first computed by computing every pixel's time course's Pearson correlation with the stimulus signal. The obtained map is then thresholded using 2 times standard deviation as a threshold before being overlaid on the mean anatomical image. Right) Correlation map showing Pearson correlation values for each pixel and the stimulus signal.

Figure 3.8 shows the case for ranks $5,3,2$ with apparent loss of information related to correlated region. Since the mean is extracted before decomposition, the main anatomical features are preserved in the mean image.

3.4.3. Preservation of Temporal Average Response using MLSVD Compression

One way to evaluate the quality of compression is through looking at the normalized mean square error that is calculated from the Frobenius norm of the difference between original and reconstructed tensor. That is;

$$NMSE_{frob} = \frac{||\boldsymbol{T} - \boldsymbol{T}_R||_F^2}{||\boldsymbol{T}||_F^2} \tag{3.7}$$

However, another measure of error calculated from the difference of average responses provide a much more well suited measure of error for the purpose of assessing preservation of functional response of the brain. Another NMSE measure is formed from the difference of these average response segments, each calculated by using an averaging window that starts at 2 seconds before and ends at 4 seconds after each stimulus for a region of interest (ROI) that includes SC region.

$$NMSE_{avg} = \frac{||\boldsymbol{s} - \boldsymbol{s}_R||^2}{||\boldsymbol{s}||^2} \tag{3.8}$$

$NMSE_{frob}$ and $NMSE_{avg}$ values computed from compression using various ranks are given in figure 3.9, with both plotted with the same colormap. As it can also be seen from Figure 3.10, $NMSE_{frob}$ values contained within a limited range fail to provide information related to how different ranks settings are performing, just providing the information that the lesser ranks we choose, the approximation becomes more erroneous as expected.

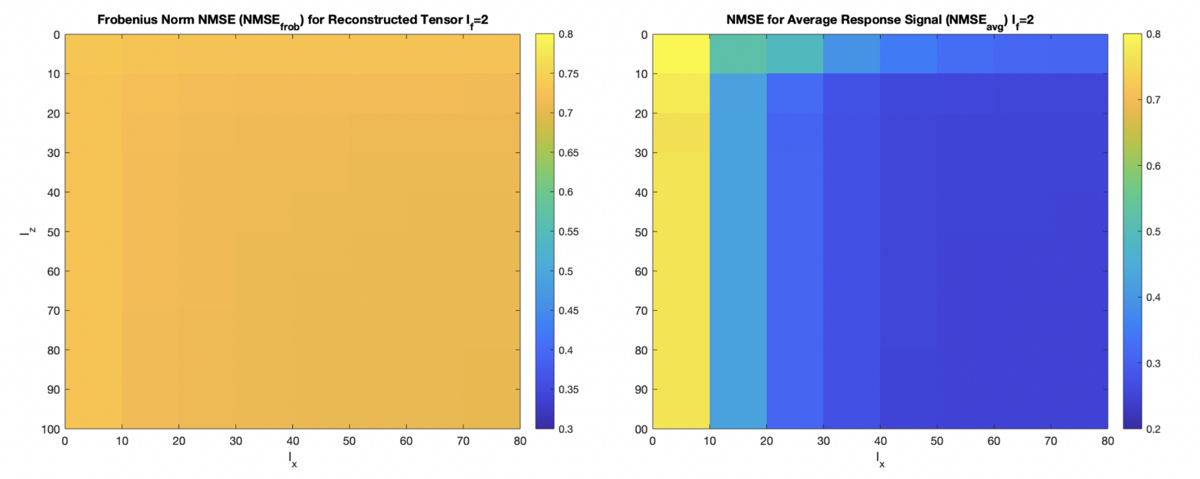


Figure 3.9. Left) Frobenius Norm Normalized Mean Square Error calculated from the difference of reconsturcted tensor and the original one, for each truncation to ranks in space for $I_f = 2$. Right) Normalized Mean Square Error calculated from the difference of average responses, for each truncation to ranks in space for $I_f = 2$. It is seen that average response NMSE provides a better comparison between the ranks in terms of the success of the compression compared to Frobenius Norm NMSE that gives similar results for each rank.

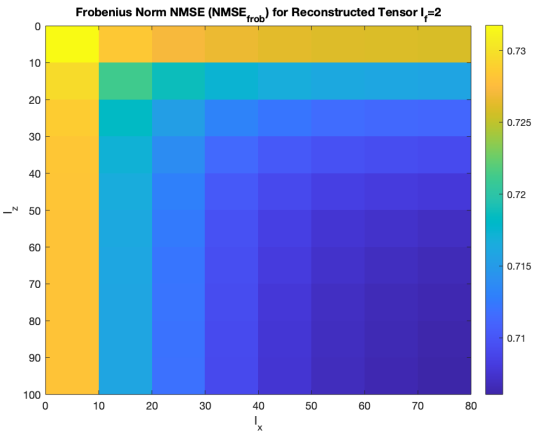


Figure 3.10. Frobenius Norm Normalized Mean Square Error values seen in different colormap. As expected, the error increases with more aggressive truncation in both modes. This doesn't provide us insight into the difference between importance of modes for truncation, or the advantage of truncating to specific ranks.

However, $NMSE_{avg}$ values provide information on which ranks and modes are more significant for preservation of the functional response of the brain, showing a clear significance of I_x dimension. If we look at each compression case, it is indeed seen that $NMSE_{avg}$ gives a valuable estimate of preservation of the functional response.

Truncation to ranks $I_z = 88, I_x = 70, I_f = 2$ whose results shown in 3.11 shows clear preservation of functional response signal. It should be noted that storage size required after this compression is 0.38% of the original data lower than that of PDIs generated by taking average of every 197 frame ensemble. On the left side, average response computed from reconstructed data is compared against the average response computed from the original data. On the right, comparison with the data using a moving average filter is made. NMSE values are given for both cases. In all cases, average responses are calculated using a bounding box surrounding the region of SC forming the region of interest (ROI), by averaging pixel's time courses that remain withing the bounding box.

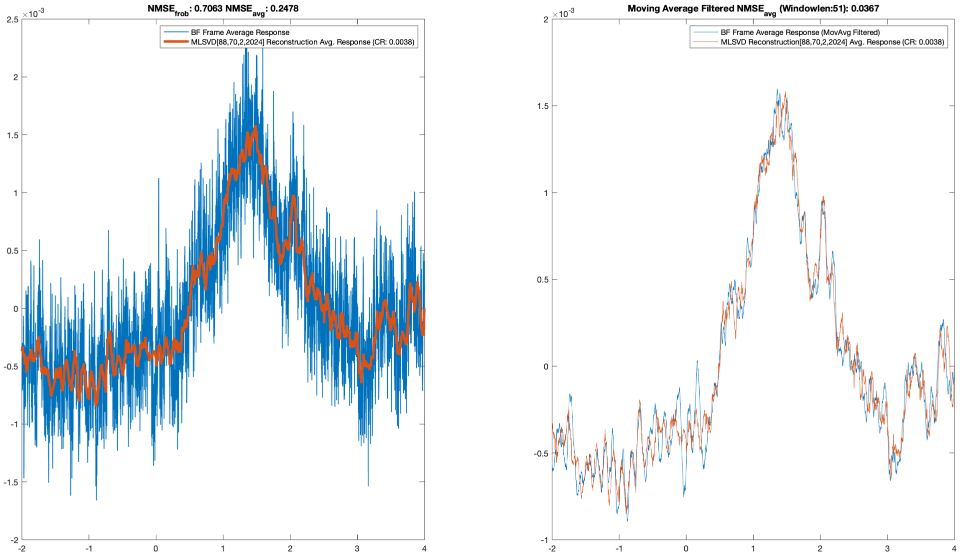


Figure 3.11. Comparison of average response calculated from ROI including SC region. Left) Average response calculated from ST-MLSVD reconstructed data with ranks [88,70,2,2024] (red) is compared against average response calculated from the original raw beamformed data with size [198,168,197,2024] (blue). Right) Average responses of ST-MLSVD reconstructed data (red) compared against average response of original raw beamformed data (blue) after both getting filtered by a moving average filter with window length 51. It is observed that average response obtained from ST-MLSVD reconstructed data follows the average response obtained from raw-beamformed data closely for these ranks, thus the average response is highly preserved.

Figure 3.12 shows the temporal characteristics of the average response signal to be preserved with ranks $I_z = 15, I_x = 65, I_f = 2$. NMSE values are similar to those seen in Figure 3.11. However one should note that correlation map calculated after this truncation had shown clear artifacts visible in Figure 3.7.

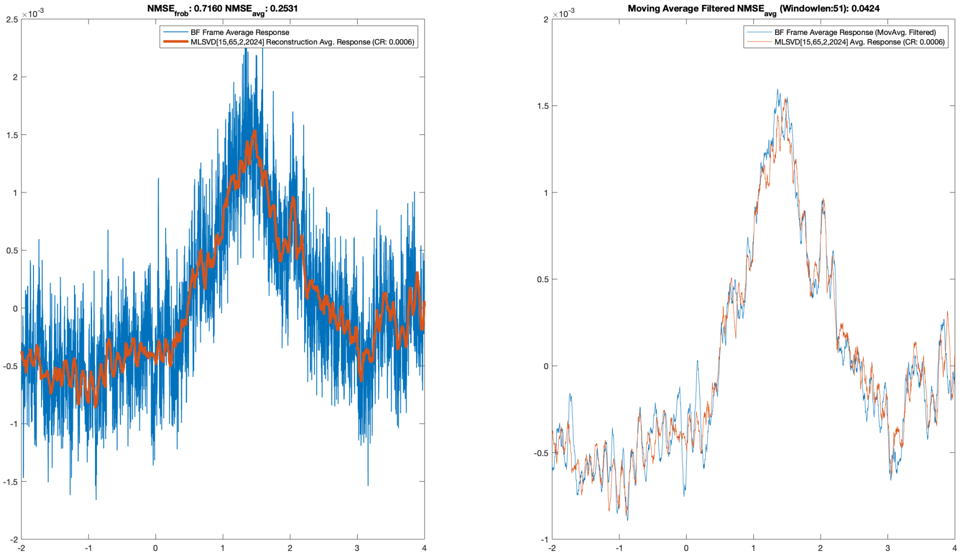


Figure 3.12. Comparison of average response calculated from ROI including SC region. Left) Average response calculated from ST-MLSVD reconstructed data with ranks [15,65,2,2024] (red) is compared against average response calculated from the original raw beamformed data with size [198,168,197,2024] (blue). Right) Average responses of ST-MLSVD reconstructed data (red) compared against average response of original raw beamformed data (blue) after both getting filtered by a moving average filter with window length 51. It is observed that average response obtained from ST-MLSVD reconstructed data follows the average response obtained from raw-beamformed data closely for these ranks, thus the average response is highly preserved as in the case of Figure 3.11

In Figure 3.13, an example of over-truncation is shown with ranks $I_z = 5, I_x = 3, I_f = 1$. The response for the reconstructed case is seen to be flat, indicating the loss of functional response. Increase in NMSE values for both cases confirm this observation.

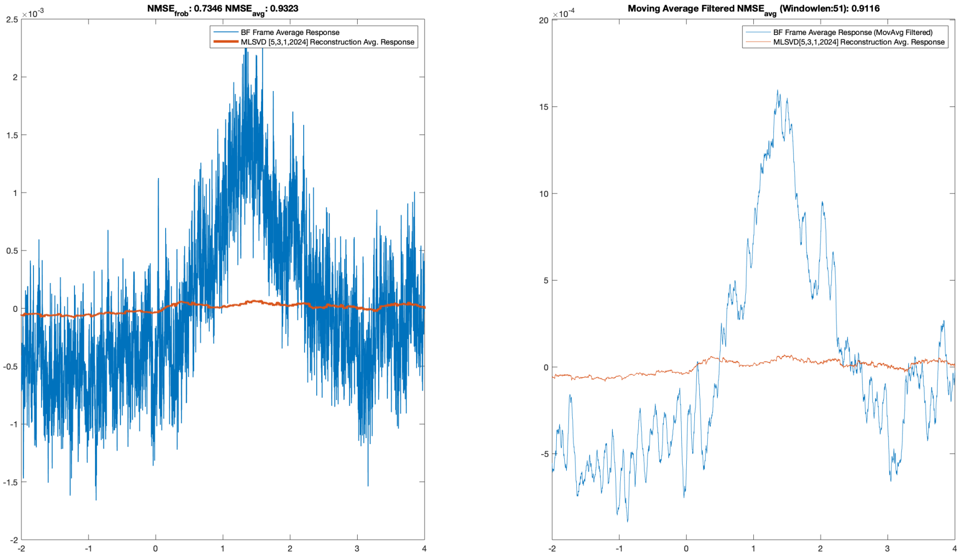


Figure 3.13. Comparison of average response calculated from ROI including SC region. Left) Average response calculated from ST-MLSVD reconstructed data with ranks [5,3,1,2024] (red) is compared against average response calculated from the original raw beamformed data with size [198,168,197,2024] (blue). Right) Average responses of ST-MLSVD reconstructed data (red) compared against average response of original raw beamformed data (blue) after both getting filtered by a moving average filter with window length 51. It is observed that the average response is not preserved in ST-MLSVD reconstructed data with such aggressive truncation with these ranks.

Figure 3.14 shows the interpolated average response calculated from the PDIs whose sampling rate is only $4Hz$. It can be seen that higher frequency characteristics that are above $4Hz$ are lost, and the exact time of the peak can only be measured at the available resolution. NMSE values are seen to be lower than in the case in Figure 3.13, but higher than 3.11 and 3.12 indicating that MLSVD compression performing better than ensemble averaging in preserving average response signal.

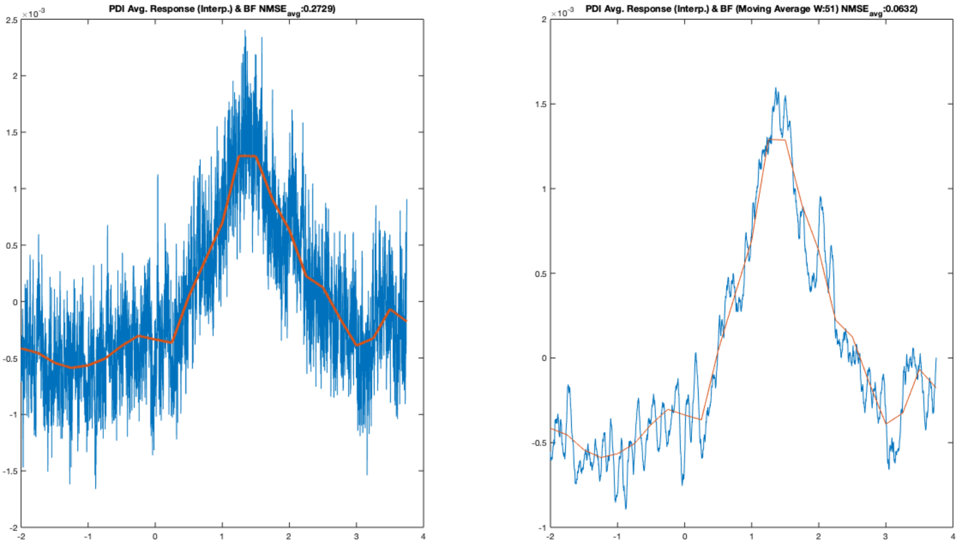


Figure 3.14. Average response calculated from PDI's compared against average response calculated from original raw beamformed data. Left) PDI's average response (red) overlaid on average response of raw beamformed data (blue). Right) PDI's average response (red) overlaid on average response of raw beamformed data (blue) after getting filtered by a moving average filter with window length 51. It is observed that although the average response peak obtained from PDI's follows the original average response obtained from raw beamformed data, higher frequency features like smaller peaks are lost, leading to a lesser resolution than the case of ST-MLSVD reconstruction presented in Figures 3.11-3.12.

In figure below; each data point corresponds to a truncation different ranks, with its normalized MSE and Compression Rate as the ratio of sizes that has to be stored in MLSVD compression and original tensors. The red circle given by PDI's compression rate and NMSE for the average is included, indicating that there are ranks that give more compression while providing less NMSE. However it should be noted that for some truncations that account for less NMSE in this plot is shown to generate a corrupted correlation map, such as in the case of truncation to spatial ranks $I_z = 15, I_x = 65$ shown in Figure 3.7

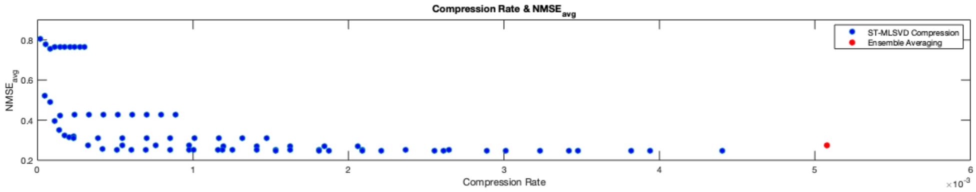


Figure 3.15. Compression rates versus normalized mean square error (NMSE) for average responses. Each blue data point corresponds to a truncation with a different rank, leading to compression of data with different rates. For each compression rate, the average response NMSE is given. The red data point indicates the compression rate and average NMSE achieved by ensemble averaging i.e. generation of PDI's. It is seen that there are ranks that provide better compression than ensemble averaging while also providing lesser average response NMSE.

3.5. Discussion

To start with, we have observed that there exists multilinear ranks that can provide an impressive decrease in the storage requirement of fUS data compared to conventional ensemble averaging. When the proposed approach is used to reconstruct the original data back from the core tensor and factor matrices, it is observed that the spatial and temporal information regarding the active region of SC is preserved by providing a correlation map similar to what the raw beamformed frames provide.

Furthermore, the thresholded overlay image shows the region of SC in a denoised manner, showing activity in only this region unlike the overlay obtained with the raw beamformed data. The denoising capability of MLSVD is also illustrated in [29].

However, truncations in spatial dimensions is also observed to lead to line-like artifacts in horizontal and vertical directions. Over-truncation leads to corruption of highly correlated regions with this artifact.

Compared to ensemble averaging at the same compression rate, ST-MLSVD is also seen to preserve temporal information of subjects' functional response to the stimulus at higher frequencies, and provide less NMSE for the average response. It should be noted that both the ST-MLSVD decomposition and computation of reconstructed data take considerably more amount of time compared to generation of PDIs.

It should also be noted that the error computed with the difference of Frobenius norm between the original and reconstructed tensor fails to provide a useful insight for preservation of functional response. From this aspect, NMSE provides a better way of assessing preservation of functional response to the stimuli. However the results should be evaluated in combination with correlation maps in order to evaluate preservation in space and time as there are cases where the temporal characteristics are preserved but information related to space is corrupted.

It is important to mention that ranks of truncation in this chapter are chosen empirically, i.e. by visual inspection of the results produced at different ranks for the dataset at hand, and for the particular slice that is being examined. A rule-based method based on multilinear singular values for automatically choosing the ranks would be ideal for this problem, providing a way to achieve a generalized compression scheme using ST-MLSVD for fUS data. Although not used in this thesis, there are attempts in the literature for automatic rank selection in MLSVD, which can be referred to in the future for extending the proposed ST-MLSVD based approach [30] [31].

4

Blind Source Separation in Multi-Slice Functional Ultrasound Data

4.1. Motivation

The brain itself is a 3-D structure, and generates activity that spreads in 3-D space. Thus, an analysis involving three dimensions of space could potentially reveal more information. This prompts us to integrate the third dimension, i.e., the slice dimension, into functional analysis pipeline.

From the previous chapter, we know that active regions in the brain appear as neighbourhoods of pixels that behave coherently in a localized fashion, as can be observed in 2-D correlation maps. The continuity of activation in 2-D space can be extended to the third dimension, as these active regions correspond to 3-D regions in reality. Thus, common information is expected to be captured in especially adjacent slices. This can be used as an a priori information in our functional analysis pipeline, allowing us to extract information more easily. Especially under noisy conditions such as the one at hand, exploiting this kind of information is expected to be valuable.

In order to form the base of multi-slice functional analysis, a model that includes multiple regions linearly mixed in time will be explored in the next section. These regions will then be attempted to be separated from each other in single slice and multi slice cases using the well-known blind source separation algorithm, Independent Component Analysis (ICA). However, as it will be seen in the model, this approach still involves vectorization, leading to the loss of spatial structure. In order to overcome this limitation, a tensor based method, Block-Term Decomposition (BTD) will be used in the following sections. BTD will be first applied to 4-D data formed by incorporating multiple slices in 3rd dimension. However, with chosen ranks and settings, results from this decomposition fails to capture all components captured by ICA. In order to overcome this, BTD will then be applied to 3-D data formed by summing adjacent slice data in order to amplify common information on them.

4.2. Signal Model Revisited

There is one particular aspect of the fUS signal related to brain activity that is ignored in the data model presented in previous chapter. That is, the existence of multiple active regions in signal related to brain activity. Instead of a single localized region, there are multiple regions of the brain corresponding to different spatial maps fluctuating with their own time courses.

$$\boldsymbol{X}_{(v)} = \boldsymbol{A}\boldsymbol{S}_{(v)} + \boldsymbol{N}_{(v)} = \sum_{r=1}^{R} a_r \times s_r^T + \boldsymbol{N}_{(v)} \tag{4.1}$$

One particular model that takes this into account is the linear mixture model [32]. Here $\boldsymbol{X}_{(v)}$ represents acquired data where each column represents a pixel's time course, rows representing acquired frames vectorized into one dimension, unfolding the data. As in the case of Equation 3.1, subscript (v) denotes vectorization operation applied, turning each image into vectors. $\boldsymbol{A}$ represents mixing matrix where each column corresponds to a time course for each row of $\boldsymbol{S}_{(v)}$. Each of these rows correspond to spatial maps vectorized into one dimension before being mixed by $\boldsymbol{A}$.

In order to incorporate multi-slice information, the data model involving two adjacent slices can be formed as below [9];

$$[\boldsymbol{X}^1_{(v)}\boldsymbol{X}^2_{(v)}] = \boldsymbol{A}[\boldsymbol{S}^1_{(v)}\boldsymbol{S}^2_{(v)}] + \boldsymbol{N}_{(v)} = \sum_{r=1}^{R} a_r \times [s^1_r s^2_r]^T + \boldsymbol{N}_{(v)} \tag{4.2}$$

In this model, matrices within square brackets correspond to concatenated matrices, doubling the data length in space. As it can be seen from the model, time courses included in $\boldsymbol{A}$ are assumed to be common for the two slices. This assumption is seen to be reasonable due to the fact that the same experimental paradigm is applied in each recording. The stimuli are applied with same delays across slices and this creates the expectation that same events will occur in the brain for us to capture in each slice.

4.3. Independent Component Analysis: Theory and Expectations

Independent component analysis (ICA) exploits independence condition in order to extract sources of a linear mixture such as presented in equation 4.1 and 4.2 [33]. Thus, it is aimed to obtain $\boldsymbol{A}$ and $\boldsymbol{S}$ in 4.1 given X which is data that is acquired from single slice. In two-slice case, $\boldsymbol{A}$ and $[\boldsymbol{S}^1_{(v)}\boldsymbol{S}^2_{(v)}]$ is aimed to be obtained given $[\boldsymbol{X}^1_{(v)}\boldsymbol{X}^2_{(v)}]$.

In this work, fastICA is used as a particular ICA algorithm that uses an approximation of negentropy, which is again and approximation to reflect probabilistic independence between spatial maps.

ICA is known to be able extract consistent and anatomically reasonable regions within fUS data as well as other neuro-imaging data, shown in previous works [34] [35] [10] [36] [37]. These regions often turn out to be task-induced, although non-task related regions are also expected to appear independently in space. Thus the first expectation related to single slice analysis is to be able to capture both task-induced and non-task related components by imposing the independence constraint.

Second, given the slices are adjacent and close to each other, consistency between components extracted from the two slices are also expected.

Third, given that the experimental paradigm is the same for both slices, the task-related regions in both slices are expected to appear in the same component for multi-slice analysis. This expectation stems from the expectation that there are commonality in time courses, which will be tested by this approach.

4.4. ICA Results

4.4.1. The Stimulus Signal

In this chapter, a canonical version of HRF is used with a longer settling duration compared to what used in previous chapter. Again, the signal formed by delayed rectangular pulses for each stimuli is convolved with the HRF, resulting in the stimulus signal shown in 4.2.

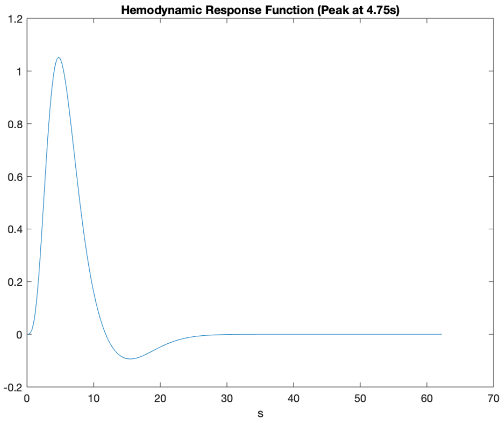


Figure 4.1. Hemodynamic Response Function (HRF) used in this chapter, peaking at 4.75 seconds.

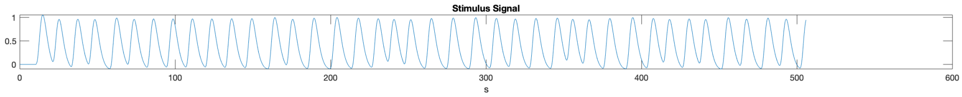


Figure 4.2. Stimulus signal formed by convolving HRF with signal formed by delayed pulses each corresponding to an air puff stimulus.

4.4.2. Pre-processing

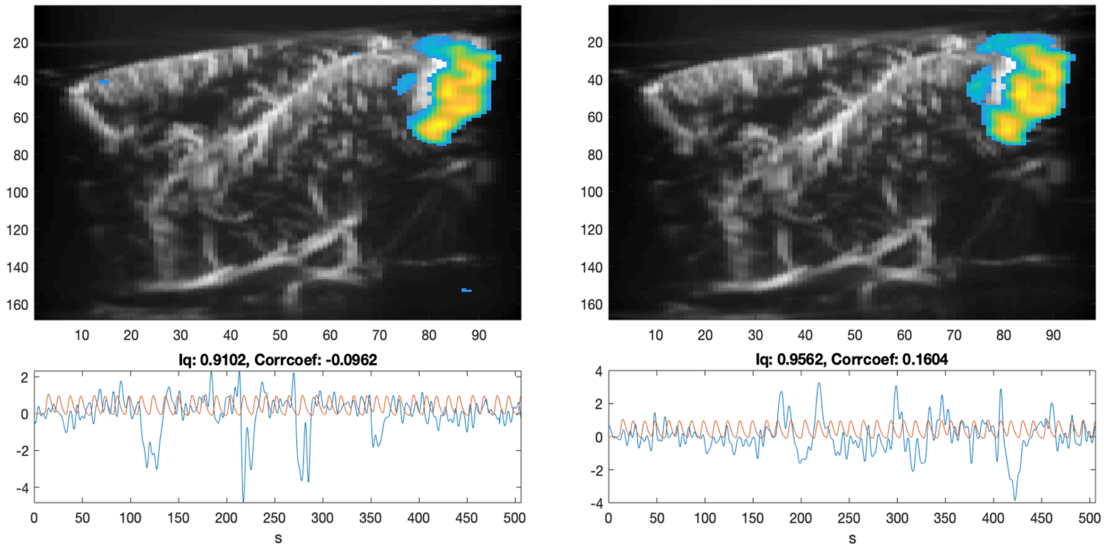
Before applying ICA, images are normalized, before being subjected to Gaussian filtering in space and time followed by low pass filtering with a butter-worth filter in time. The kernel size of the Gaussian filter is 5 by 5 in space and 7 pixels in time with $\sigma = 2$ for all dimensions. Cut-off frequency of Butterworth filter is chosen at 0.2 Hz.

4.4.3. Single Slice

Spatial maps that are obtained from single slice ICA and their time courses are shown in Figure 4.3-4.6. In total, 20 components are extracted 4 of which are observed to be corresponding to anatomical regions. These regions are confirmed to be corresponding to anatomical regions by finding them in The Allen Mouse Brain Atlas [38] as well as looking at the binary masks produced in standardized results provided by Center for Ultrasound and Brain imaging at Erasmus Medical Center (CUBE) which also have provided the data.

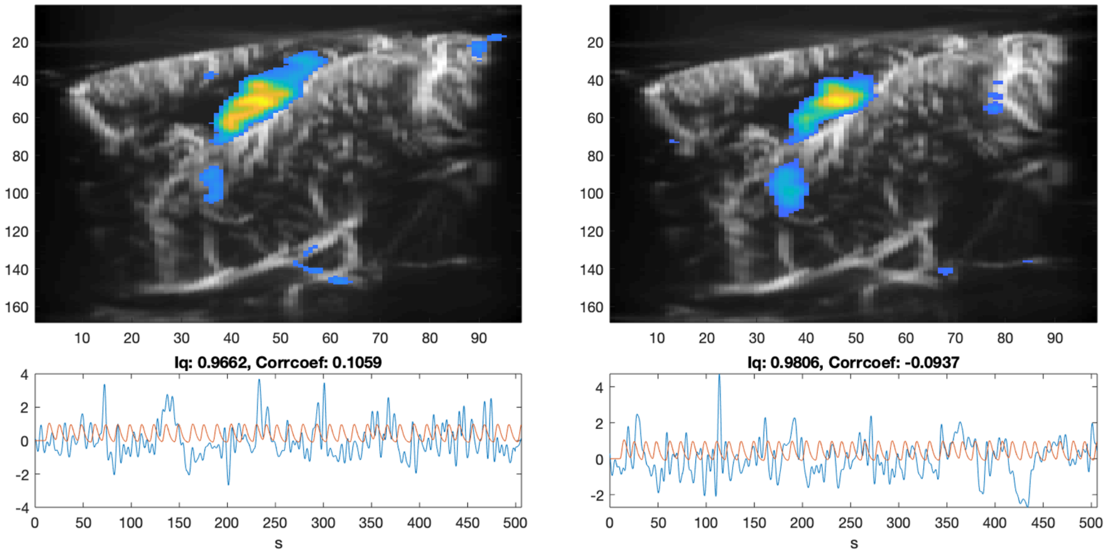
Each of these figures have a stability index denoted by Iq, and a correlation coefficient denoted by corrcoef. The stability index Iq is computed by the ICASSO package on Matlab which allows for assessing stability of each component over multiple trials as follows. In ICASSO framework, fastICA is run multiple times, obtaining an estimate for de-mixing matrix W for each trial which corresponds to pseudo-inverse of A given in Equation 4.1. The columns of these de-mixing matrices are clustered, and the centroid of each cluster is determined as the cluster estimate, forming a new, reliable W. Using this W, the cluster estimated mixing matrix A and cluster estimated spatial maps S is computed. Since each spatial map that forms the rows of the new S is a representative of a cluster of trials, the Iq value evaluates the compactness and orthogonality of the cluster, giving an estimate of the components' stability. In ideal case, Iq is given as 1 [39]. Thus each result shown in Figures 4.3-4.6 is representative of a cluster of results from multiple fastICA runs. Corrcoef is computed as the Pearson correlation between the time course of the component and the stimulus signal, given in Figure 4.2.

It is observed that all 4 components that correspond to anatomical regions are common in both slices, although only the region Superior Colliculus (SC) has a time course significantly correlated with the stimulus signal in both slices. Thus as the only stimulus related component, it is expected that the assumption of common time courses between slices will only work for this region.



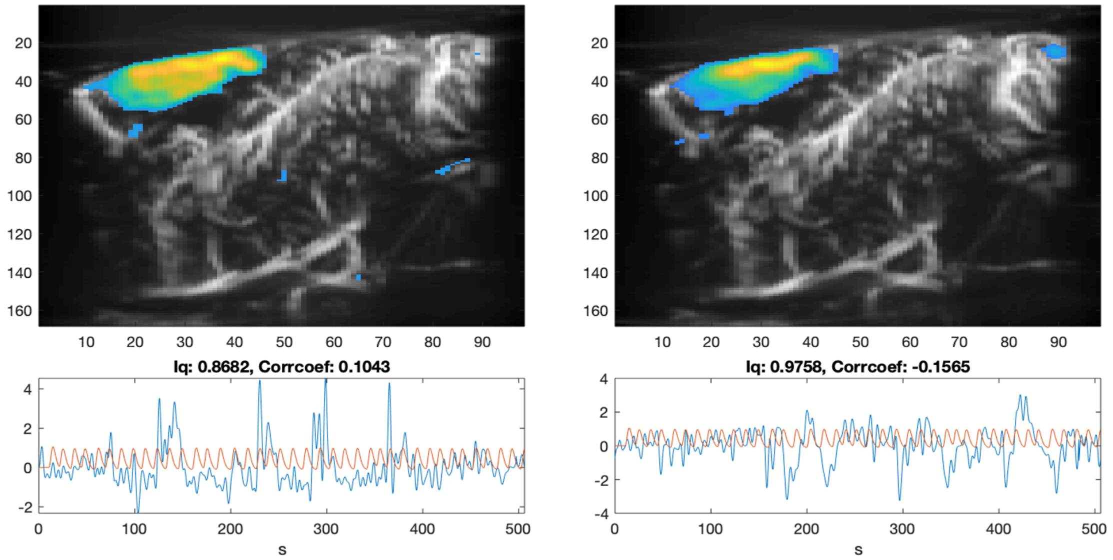
a) b)

Figure 4.3. Cerebellum (CR) region, a) Slice 25, b) Slice 26. Spatial maps are thresholded using 2 times standard deviation. Yellow corresponds to higher intensity response while blue corresponds to lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.



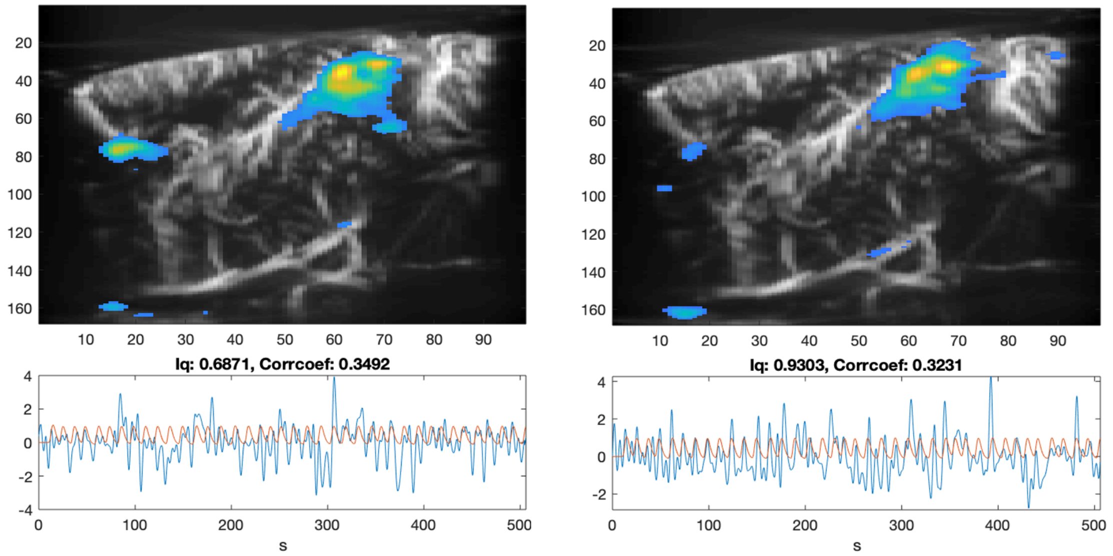
a) b)

Figure 4.4. Spatial maps including Hippocampus (HC) region, a) Slice 25, b) Slice 26. Spatial maps are thresholded using 2 times standard deviation. Yellow corresponds to higher intensity response while blue corresponds to lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.



a) b)

Figure 4.5. Spatial maps including Motor Cortex (MC) region, a) Slice 25, b) Slice 26. Spatial maps are thresholded using 2 times standard deviation. Yellow corresponds to higher intensity response while blue corresponds to lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.



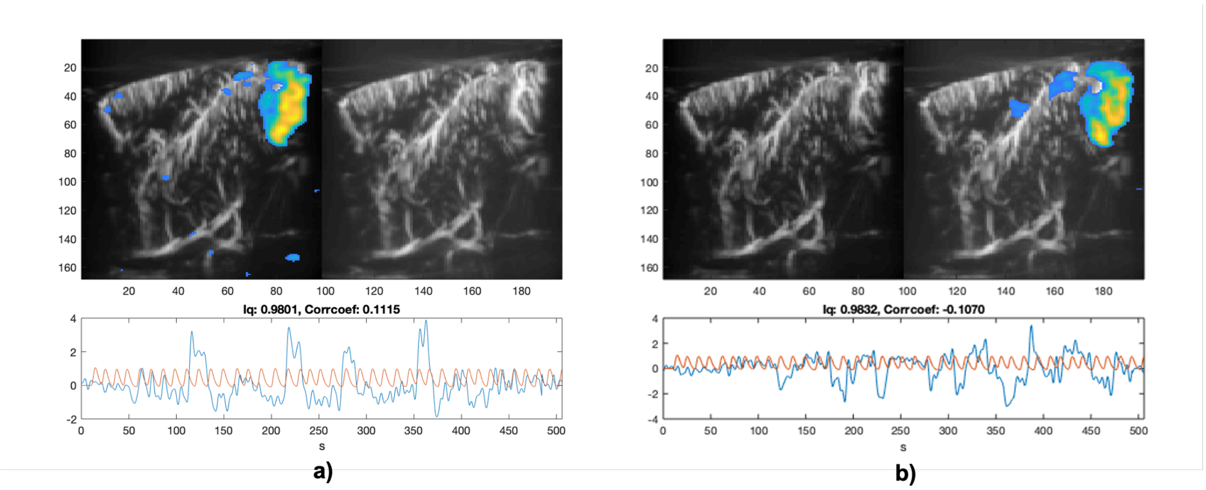
a) b)

Figure 4.6. Superior Colliculus (SC) region, a) Slice 25, b) Slice 26. Spatial maps are thresholded using 2 times standard deviation. Yellow-Orange means higher intensity response while Blue means lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.

As expected, anatomically reasonable regions are captured consistently in both slices. Furthermore, it is shown that there is more happening in the data than task-induced activation, given the only task-related component turned out to be SC. Task-related component here refers to the component that has a highly correlated time course with the stimulus signal given in Figure 4.2.

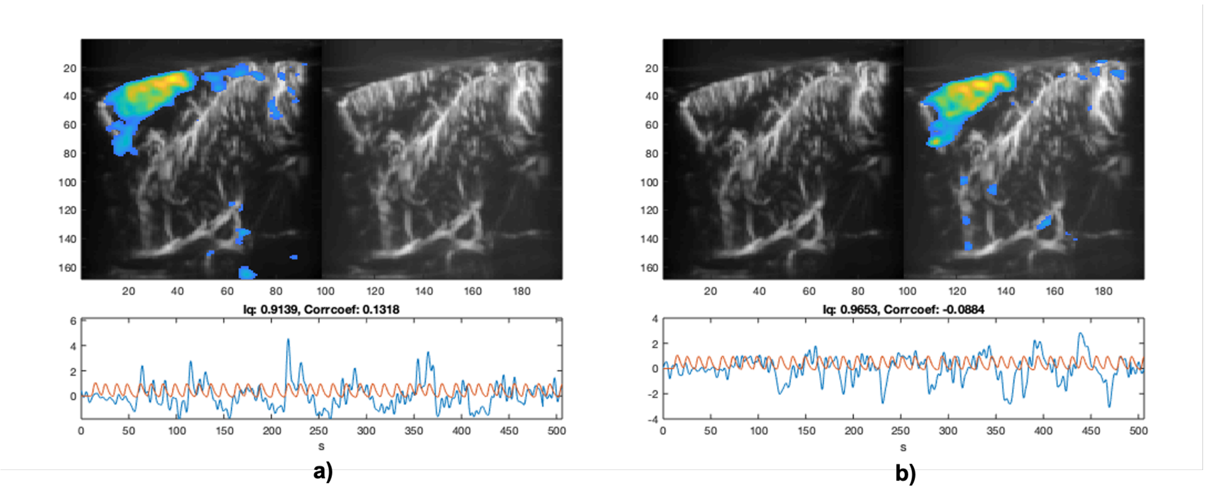
4.4.4. Multi-Slice

In order to process slices simultanously, the two slices are concatenated in space after going through preprocessing as shown in 4.2. As in the case of single slice results, ICASSO package is used to produce results shown in Figures 4.7-4.9, giving a stability index for each component.



a) b)

Figure 4.7. a)Cerebellum (CR) appearing on Slice 25, b) Cerebellum appearing on Slice 26. Spatial maps are thresholded using 2 times standard deviation. Yellow corresponds to higher intensity response while blue corresponds to lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.



a) b)

Figure 4.8. a)Motor Cortex (MC) appearing on Slice 25, b)MC appearing on Slice 26. Spatial maps are thresholded using 2 times standard deviation. Yellow corresponds to higher intensity response while blue corresponds to lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.

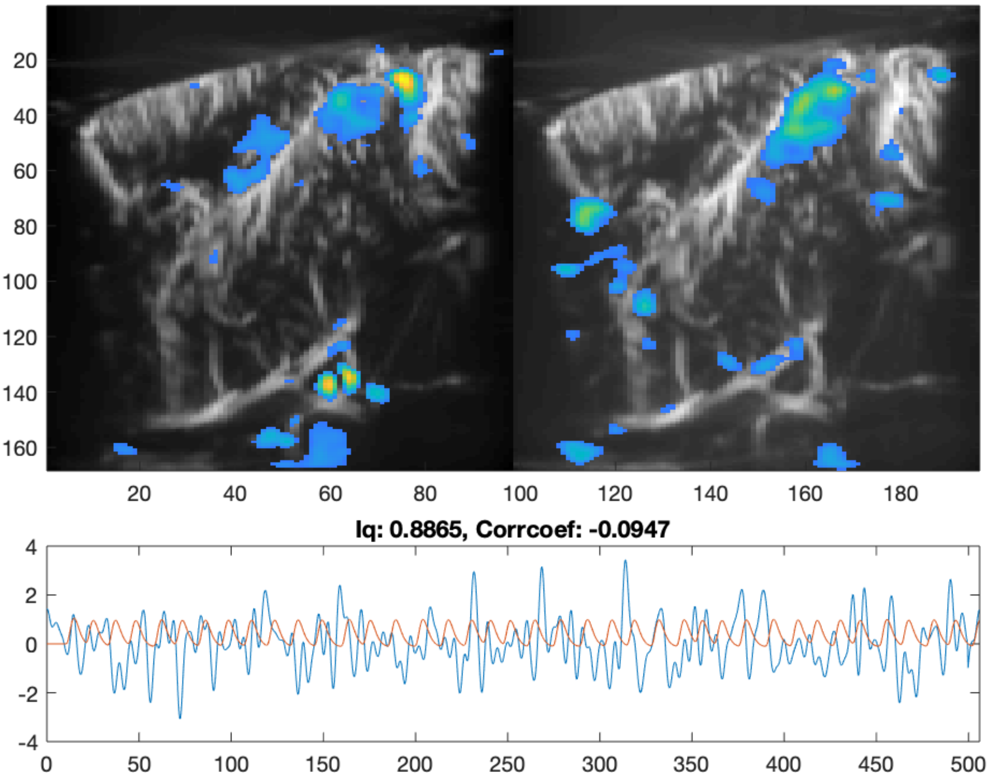


Figure 4.9. Superior Colliculus (SC) appearing in both slices in same component. Spatial maps are thresholded using 2 times standard deviation. Yellow corresponds to higher intensity response while blue corresponds to lower intensity response. Iq denotes stability index computed by ICASSO package after clustering multiple trials. Corrcoef indicates Pearson correlation of the time course with the stimulus signal.

It is observed that the only region that appeared in both left and right slices in one component is the region SC, which is known to be the task related region that is being activated in correlation with the stimulus in each slice. However in this multi-slice case high correlation information seems to be lost indicating temporal differences between slices despite the spatial commonality. A highly probable reason for this is that the slices are recorded at different time instances, although recorded in sequential manner. The 2-D probe can only record one slice at a time, leading to these differences in time courses between slices. Thus, as seen from the results presented here, the assumption of common time courses between slices for the task related regions are shown to be invalid.

A reasonable candidate for the temporal differences between two slices is the delay of response to the stimuli. Although in both components correlated time courses are captured, they might have different delays to the stimuli that makes each of the time course differ.

4.4.5. Limitations

The limitation of not incorporating 3-D information into the functional analysis pipeline is evident in single slice ICA, which includes processing of slices separately. This is overcome by multi-slice ICA where slices are concatenated in space before given as inputs to the algorithm. However, this method still involves vectorization of images.

In both ICA approaches, vectorization is applied to each frame before the decomposition. This leads to the loss of spatial structure which can be utilized by a low rank model in higher order to extract useful information. A simple way to illustrate this is to think about two points that are very close to each other on an image, forming a larger area that is known to be active. This larger area consisting of two points can be expressed by outer product of a small number of vectors, leading to a low rank form. However when images are vectorized, points that form this larger area end up in further away places, preventing us from utilizing such structure in data [25].

4.5. Signal Model Revisited Again

In order to provide a mixture model that avoids vectorization, the following 4-D model of the multi-slice data where $\boldsymbol{X}_m^{1,2} \in R^{N_z \text{ x } N_x \text{ x } N_y \text{x} N_F}$ where $N_y = 2$ is presented.

$$\begin{aligned} \boldsymbol{X}_m^{1,2} = \sum_{r=1}^{R} \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 U_y^r \times_4 U_f^r + \boldsymbol{N} = \sum_{r=1}^{R} \boldsymbol{S}^r \times_4 U_f^r + \boldsymbol{N} \\ \text{where } \boldsymbol{S}^r = \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 U_y^r \end{aligned} \tag{4.3}$$

In this representation, each U_r correspond to a factor matrix, whose columns represent signatures that represent information related to the underlying sources in their respective dimensions across time and space. For each component, $\boldsymbol{G}^r$ is a tensor that determines interactions between these factor matrices in their modes. These interactions are enforced by the n-mode product between the core tensors and factor matrices denoted with $\boldsymbol{G}^r \times_n U^r$.

Here, $\boldsymbol{G}^r$ corresponds to core tensors of each component while U_z^r, U_x^r correspond to factor matrices for the first two dimensions. The factor matrices related to the slice dimension are denoted by U_y^r whereas the factor matrices related to time are denoted by U_f^r.

A comparison made with model presented in Section 4.2 reveals the connection between the two models. Here, spatial maps from each slice is contained in $\boldsymbol{S}^r$ for each component with index r. U_f^r take the same role with a_r described in Equation 4.1 in this sense, containing time courses that span spatial maps onto a new dimension. This analogy between the two models is also utilized in [40] from the other way around by segmenting an initially two-dimensional data modeled as in Equation 4.1 and applying BTD to the newly formed tensor.

A simpler model that could be used for representing two slice information without vectorization can be the one presented below, where $\boldsymbol{X}_m^{1,2} \in R^{N_z \text{ x } N_x \text{x} N_F} = \boldsymbol{X}_m^1 + \boldsymbol{X}_m^2$;

$$\begin{aligned} \boldsymbol{X}_m^{1,2} = \sum_{r=1}^{R} \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 U_f^r + \boldsymbol{N} = \sum_{r=1}^{R} \boldsymbol{S}^r \times_3 U_f^r + \boldsymbol{N} \\ \text{where } \boldsymbol{S}^r = \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \end{aligned} \tag{4.4}$$

Here, the two slices are summed to form $\boldsymbol{X}_m^{1,2}$, and as in the case of 4.3, $\boldsymbol{G}^r$ corresponds to core tensors of each component while U_z^r, U_x^r correspond to factor matrices for the first two dimensions. Again, U_f^r corresponds to factor matrices for the time dimension.

The motivation behind the model presented in 4.4 is to amplify the common active regions in both slices while suppressing noise which is assumed to be uncorrelated across space, time and slice dimensions.

In the following sections, models that are derived from either 4.3 or 4.4 will be used.

4.6. Block Term Decomposition: Theory and Expectations

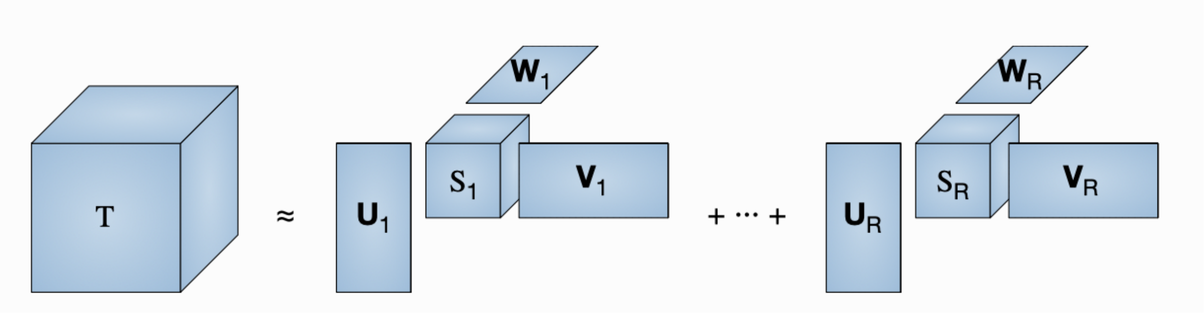


Figure 4.10. Illustration of Block term decomposition on a third order tensor

In order to extract core tensors and factor matrices illustrated in 4.3 and 4.4, Block Term Decomposition implemented in Tensorlab package in Matlab will be used. Unlike ICA BTD does not use the constraint of independence [41]. The problem at hand comes down to the following optimization problem solved via nonlinear least squares approach [42].

$$\min_{U_1,U_2,\ldots,U_N} \quad ||\sum_{r=1}^{R} \boldsymbol{G}^r \times_1 U_1^r \times_2 U_2^r, \ldots, \times_N U_N^r - \boldsymbol{X}_m^{1,2}||^2 \tag{4.5}$$

The expectation is that preserving the spatial structure will enable us to extract spatial maps easier under noisy conditions, leading the algorithm to find the true solution easier. The term easy here includes finding a solution under milder uniqueness conditions as opposed to those enforced in matrix-based analysis such as independence of sources in ICA [43].

Another expectation is that the regions already shown in the results of ICA will be extracted, allowing for them to be expressed in low-rank form, indicating utilization of their spatial structure in the data.

4.7. BTD Results

This section will consist of three parts. First, a simulation that illustrates the blind source separation capability of BTD using two stock images will be presented. Next, decomposition of 4-D data formed by concatenating pixels on a new dimension will be done, extracting anatomically reasonable regions from the data as in the case of ICA. As an alternative approach, two slices will be summed to form a 3-D aggregate data, and this sum will be decomposed with the goal of extracting common information between the slices. The results will be evaluated and scored using a binary mask of the region SC obtained from ICA trials.

4.7.1. Simulation

In this section, a simulation using two images is performed in order to illustrate the power of BTD in blind source separation.

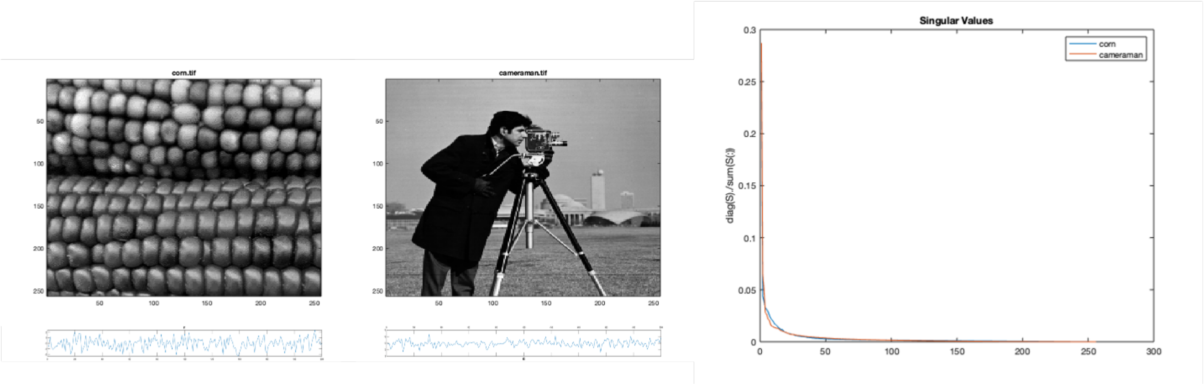


Figure 4.11. Corn and cameraman images are given with their respective time courses generated by them, simulating brain regions that are stable in space, fluctuating in time. On the right, distribution of singular values for each image treating them as matrices to do SVD on is found. The values are mostly concentrated below the rank of 50.

Figure 4.11 shows the two images that are used in this simulation, along with the distribution of singular values resulting from the SVD applied on images. The majority of singular values for both images turn out to be included under rank 50. Thus we will approximate the images with rank 50 first.

The time courses shown below the images are generated randomly as zero-mean for each photograph, with a variance of 1.

Figure 4.12. Two mixtures of images each consisting of corn and cameraman images representing each slice is presented with their respective time courses. Both cameraman and the corn image is zoomed in and a moved portion of the image is cropped to generate different versions that would be included in different slices.

Using the modified versions of two images, a two-slice data is formed in $R^{N_z \text{ x } N_x \text{ x } N_y \text{x} N_F}$ as shown in 4.12. This modification is done by zooming into the full images and cropping them differently, leading to "moved" details to appear in each of the two cases.

By taking the outer product of images and time courses, sets of images that vary in time in terms of intensity values are formed. Then, the corn image set is added onto cameraman image set, creating a mixture for each slice similar to the case of real data.

Here, a modified version of the data model presented in 4.3 is used;

$$\begin{aligned} \boldsymbol{X}_m^{1,2} = \sum_{r=1}^{R} \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 U_y^r \times_4 u_f^r + \boldsymbol{N} = \sum_{r=1}^{R} \boldsymbol{S}^r \times_4 u_f^r + \boldsymbol{N} \\ \text{where } \boldsymbol{S}^r = \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 U_y^r \end{aligned} \tag{4.6}$$

where the factor matrices corresponding to the time dimension are actually rank 1, leading the extracted images to be stable in space, scaled by a constant for each time point. For the rank of slice dimension factor matrix U_y^r, 2 is chosen leading to a BTD with ranks L,L,2,1.

In order to evaluate the quality of separation, a metric defined in [44] is used. It is called Absolute Concrete Correlation Distance (ACCD), presented below;

$$\begin{aligned} \mathrm{ACCD}_{i,j} = |\rho(\boldsymbol{A}_{o_i}, \boldsymbol{A}_j)| - |\rho(\boldsymbol{A}_i, \boldsymbol{A}_j)| + 1 \\ i, j = 1, 2, ..., R. \end{aligned} \tag{4.7}$$

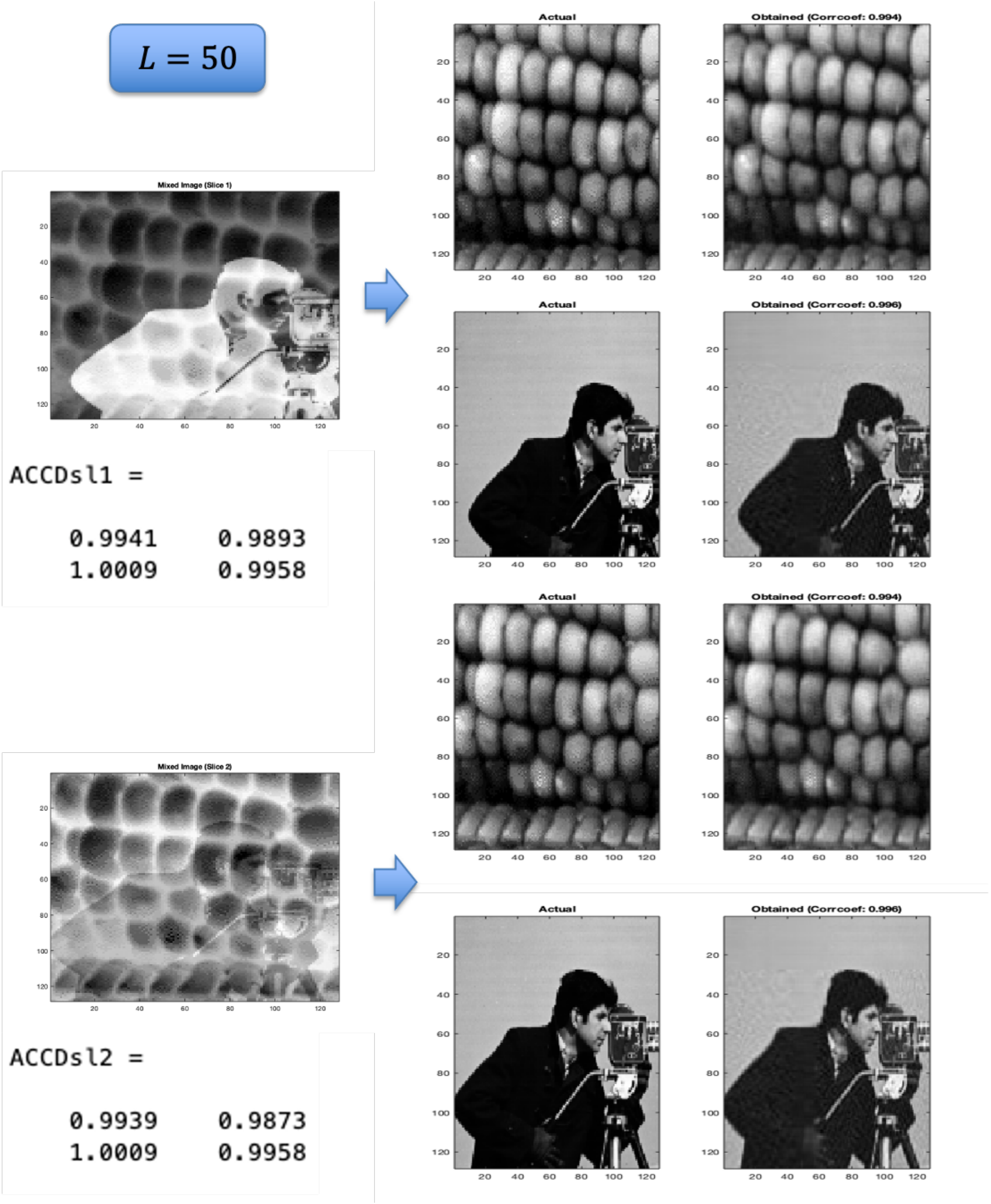
Here, $\boldsymbol{A}_o$ denotes obtained sources whereas $\boldsymbol{A}$ denotes actual sources. $\rho()$ is used to denote Pearson correlation. The indices i and j here denote each image, matched to these indices whether they are obtained or actual. In our case, the index i corresponds to the cameraman image whereas index j corresponds to the corn image.

Using these indices, a matrix is defined, formed by scores obtained from the expression. For $i = j$ the expression reduces to the Pearson correlation between the obtained image and actual image, forming the diagonal values of the matrix. For these values, the values range between 0 to 1 as expected, 1 being a perfect match between the obtained image and acquired image.

However, as reported in [44], these diagonal values do not provide the full picture for the evaluation of separation. For $i \neq j$, The metric evaluates cross-talk between obtained components, indicated in off-diagonal values. In a perfect separation, the first term indicating correlation between an obtained image and the other actual image is canceled by the correlation between actual images themselves, leading to the value of 1 for the expression. If there's cross-talk between obtained image and the other actual image, i.e. if theres a change in how the images are similar to each other after the separation, the first term differs from the second term, leading the value to deviate from 1.

Thus, a perfect separation results in a matrix consisting of all 1's. Deviation from 1's in diagonal values indicate that the images are not extracted properly, for instance there is a mismatch between obtained corn image and the actual corn image. Deviation from 1's in off diagonal values indicate that there is cross-talk between images, for instance the obtained corn image is more similar to the actual cameraman image than the actual corn image. Ideally, these to similarities measured by correlation should be the same.

Results of setting L to two different ranks are shown below in Figure 4.13 and Figure 4.14. It is observed that although the true rank is around 50, setting lower values to L also leads to good results, indicating that lower sensitivity to the choice of ranks for this problem.



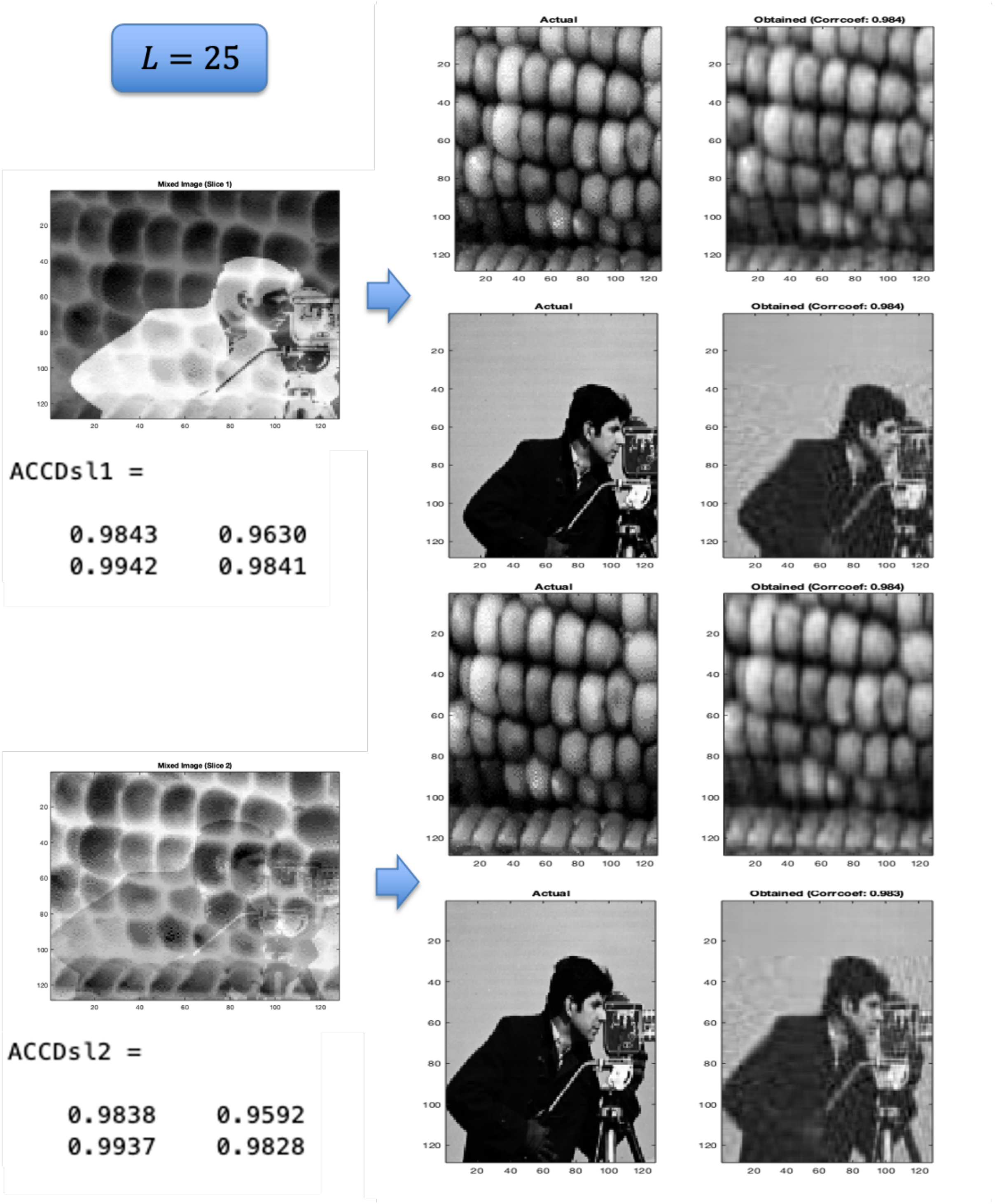
```
ACCDsl1 =

    0.9941    0.9893
    1.0009    0.9958
```

```
ACCDsl2 =

    0.9939    0.9873
    1.0009    0.9958
```

Figure 4.13. Simulation results when multilinear rank L is set to 50. On the left, mixed image captured at first time instance of the mixtures of first slice (top) and second slice (bottom) is shown. On the right, result of the BTD separation is shown with obtained images from the mixture in comparison with actual images. ACCD matrix for evaluating the quality of separation is given under each mixed image on the left.



```
ACCDsl1 =

    0.9843    0.9630
    0.9942    0.9841
```

```
ACCDsl2 =

    0.9838    0.9592
    0.9937    0.9828
```

Figure 4.14. Simulation results when multilinear rank L is set to 25. On the left, mixed image captured at first time instance of the mixtures of first slice (top) and second slice (bottom) is shown. On the right, result of the BTD separation is shown with obtained images from the mixture in comparison with actual images. ACCD matrix for evaluating the quality of separation is given under each mixed image on the left.

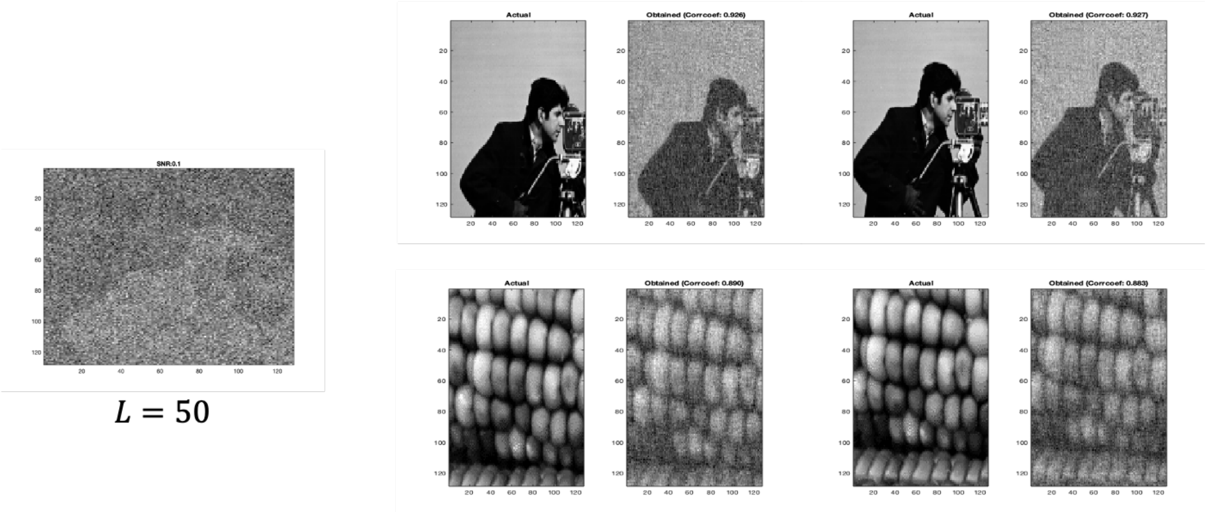


Figure 4.15. Performance with Gaussian noise added to the mixture. Mixture image captured at the first time instance is given on the left. In the middle, obtained images from slice 1 is shown in comparison to actual images. On the right, obtained images from slice 2 is shown in comparison to actual images.

In Figure 4.15, de-noising capability of BTD is shown by adding Gaussian noise to the mixtures, still leading to a successful separation for two slices. A successful application of BTD on noisy data is included in [45].

4.7.2. 4-D Decomposition

In this section, sets of images formed by using a variation of the decomposition model used in 4.3 is used, by setting factor matrices related to slice and time dimensions to rank-1 as shown in 4.8. Underlying assumption for this model is that regions that form each source are stable in space, but their activity is fluctuating in time. Thus for each source, an image corresponding to a spatial map whose activity is varying in time is extracted.

The results shown here are obtained after running ST-MLSVD on the data with core-tensor dimensions $90, 80, 2, 30$ denoted by $\boldsymbol{S}^{1,2}$ with ranks related to space chosen similarly to those that produced successful results in Chapter 3. This step is seen necessary to reduce computation time, compressing the data. After obtaining the core tensor, $L, L, 1, 1$ Block-Term Decomposition is applied to the data. This pipeline is similar to the ICA pipeline which involves PCA as a compression step before decomposing the data. For BTD, L is set to 30 with $R = 10$ whereas random initialization is used. All results in this section are obtained by running BTD for a single trial with the given parameters.

$$\begin{aligned} \boldsymbol{S}^{1,2} = \sum_{r=1}^{R} \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 u_y^r \times_4 u_f^r + \boldsymbol{N} = \sum_{r=1}^{R} \boldsymbol{S}^r \times_4 u_f^r + \boldsymbol{N} \\ \text{where } \boldsymbol{S}^r = \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 u_y^r \end{aligned} \tag{4.8}$$

These images are expected to correspond to spatial maps of active regions as in the case of ICA.

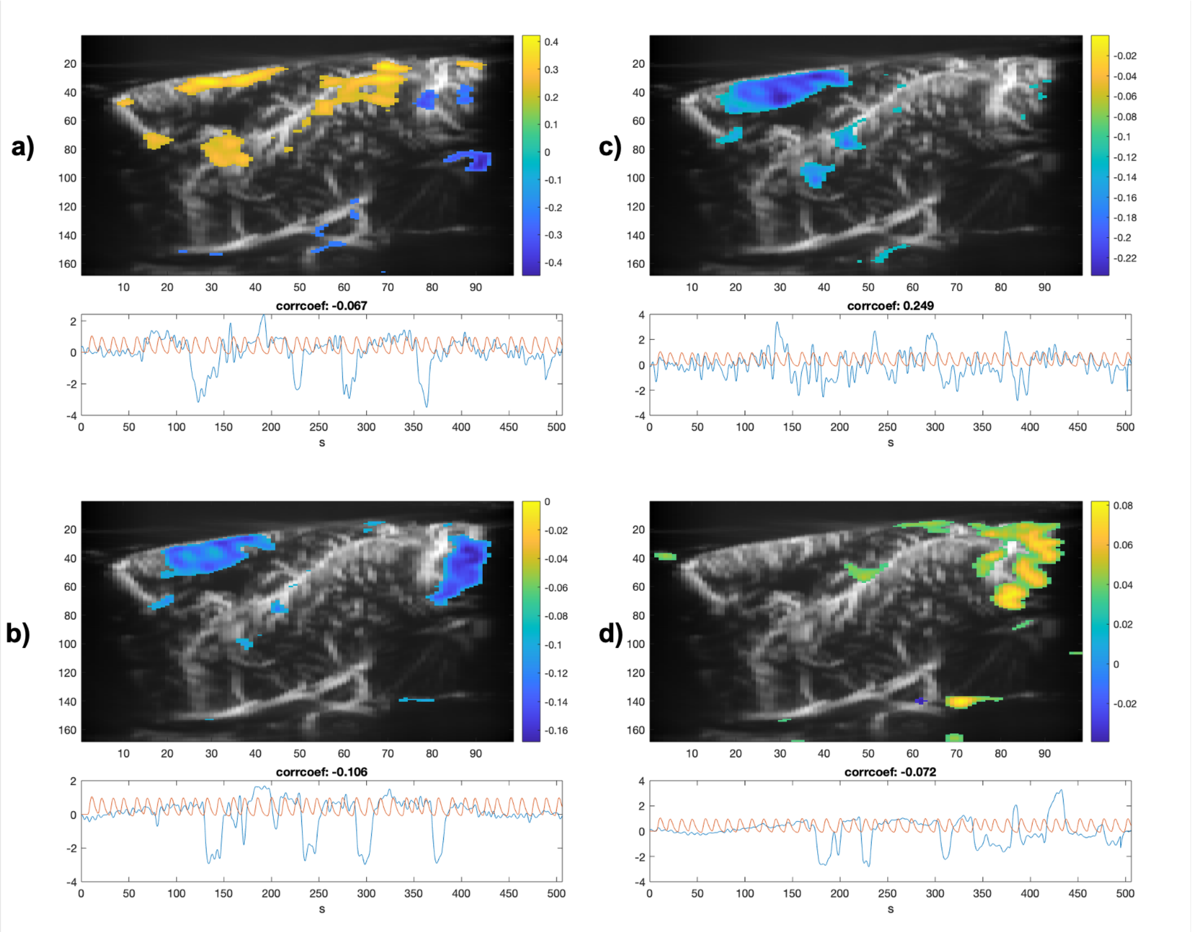


Figure 4.16. a) Spatial map partially including SC region. b) Spatial map including MC and Cerebellum regions. c) Spatial map including MC region. d) Spatial map including Cerebellum region. Corrcoef denotes Pearson correlation of the time courses with the stimulus signal. Spatial maps are thresholded with 2 times standard deviation for each image before being overlaid on the mean PDI of Slice 25.

The 4 spatial maps shown in Figure 4.16 are chosen as the anatomically valid regions among 10 spatial maps that are extracted. The selection is made by comparing spatial maps to binary masks produced in standardized results by CUBE as well as looking at the corresponding regions in Allen Mouse Brain Atlas,

It is observed that although anatomically valid regions are apparent in these results, the method fails to extract and separate all components that are extracted by ICA, including the task-induced active region of SC. Extracted time courses produce low correlation values with stimulus signal, indicating low performance of the algorithm compared to single slice ICA which was able to achieve higher correlations for SC region. However, it is able to capture relatively higher correlations for the regionMC, for which ICA had failed to capture. This indicates that this region may also be regarded as a task-induced region, providing a correlated response to the stimulus. Indeed in [46], it is mentioned that activity in MC is necessary for coordination of a response to the air-puff stimulus.

Trials using the rank of $R = 5,20$ and multi-linear rank of $L = 10,50$ produced similar results with the region MC and Cerebellum appearing in spatial maps, indicating the sufficiency of these ranks for extracting these regions. However higher correlation withMC was only observed again when $R = 10, L = 50$ showing sensitivity to changes in R.

4.7.3. Mixture Decomposition

In this subsection, Equation 4.4 is used as a model for the decomposition where $\boldsymbol{X}^{1,2}$ is formed as;

$$\boldsymbol{X}^{1,2} = \boldsymbol{X}^1 + \boldsymbol{X}^2 \tag{4.9}$$

The assumption is that by setting $R = 3$, three components will be extracted. These are assumed to be consisting of the component that correspond to common information in two slices, the component of uncommon information in slice 1 and the component of uncommon information in slice 2.

$$\begin{aligned} \boldsymbol{X}^{1,2} = \sum_{r=1}^{R} \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \times_3 U_f^r + \boldsymbol{N} = \sum_{r=1}^{R} \boldsymbol{S}^r \times_3 U_f^r + \boldsymbol{N} = \boldsymbol{S}_{common} + \boldsymbol{N}^1 + \boldsymbol{N}^2 \\ \text{where } \boldsymbol{S}^r = \boldsymbol{G}^r \times_1 U_z^r \times_2 U_x^r \end{aligned} \tag{4.10}$$

The motivation behind this model is based on the following assumption. The task related component of SC is too weak compared to noise for BTD to extract from single slice in both slices. By adding two slices the common information is amplified while the noise is suppressed, making it easier to extract the SC.

Since the factor matrix related to time is not rank-1 anymore, we are able to extract images changing in time for each component as opposed to the spatially stable images that only fluctuate in time as a result of Block-Term Decomposition with L,L,M ranks for each factor matrix. The main ranks used to produce results presented here are $L = 20, M = 10$. This choice of relatively low ranks is enforced by time constraints as each decomposition presented here are run 100 times before picking the best candidate trial to examine.

The sets of images acquired from the decomposition for each component are correlated against the stimulus signal pixel by pixel. The correlation maps are then thresholded adaptively using 2 times standard deviation and evaluated using the mask of SC region shown in 4.17.

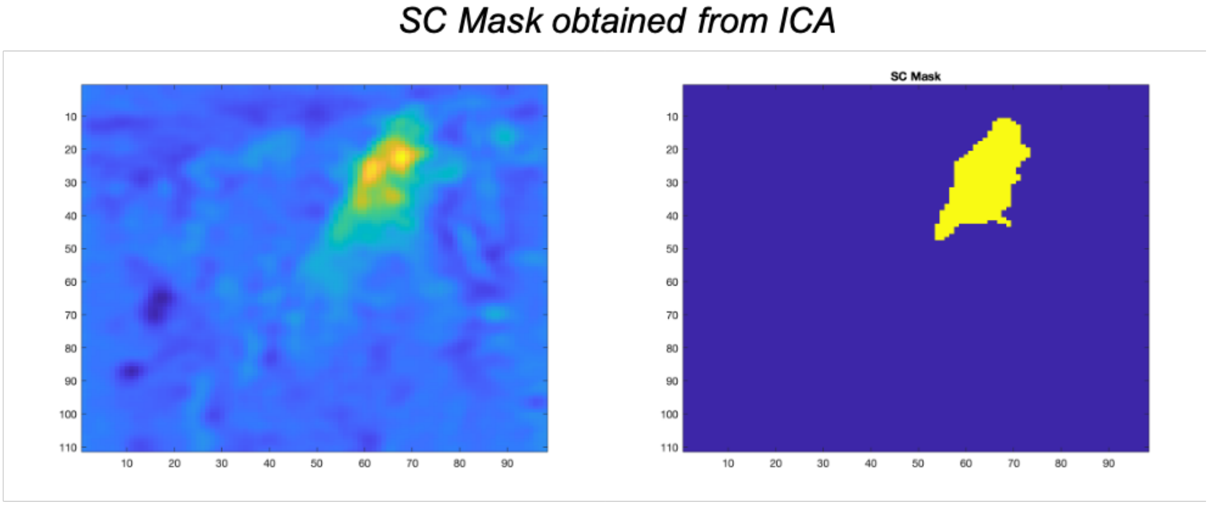


Figure 4.17. Binary mask of the SC region obtained from ICA results. Left) Spatial map obtained as a component in ICA. Right) Binary mask obtained using the threshold of 2 times standard deviation.

Using the mask, the ratio of pixels that remain within the region are computed along with the ratio of those that remain outside. In Figure 4.18 result of the best trial of decomposition on the whole two-slice data is given, chosen among 100 trial runs of the algorithm. The choice of the best trial is done by looking at the final score, calculated by subtraction of ratio of pixels that remain outside the mask from the ratio of those that remain inside. The final score for each component extracted is calculated as;

$$\text{Final Score} = \frac{\#(C_{\text{map}} \cap SC_{\text{mask}}) - \#(C_{\text{map}} \cap -SC_{\text{mask}})}{\#(C_{\text{map}} \cap SC_{\text{mask}}) + \#(C_{\text{map}} \cap -SC_{\text{mask}})} \tag{4.11}$$

Where C_{map} denotes thresholded correlation map corresponding to a binary mask. SC_{mask} corresponds to the binary mask of SC obtained from ICA results. Operator # denotes number of pixels.

A perfect score is 1, where all pixels of the thresholded correlation map remain within the binary mask. The worst score is -1 where all pixels of the thresholded correlation map remain outside of the binary mask. Higher scores are in this sense better, conveying that more of the thresholded correlation map remains within the binary mask compared to those remain outside.

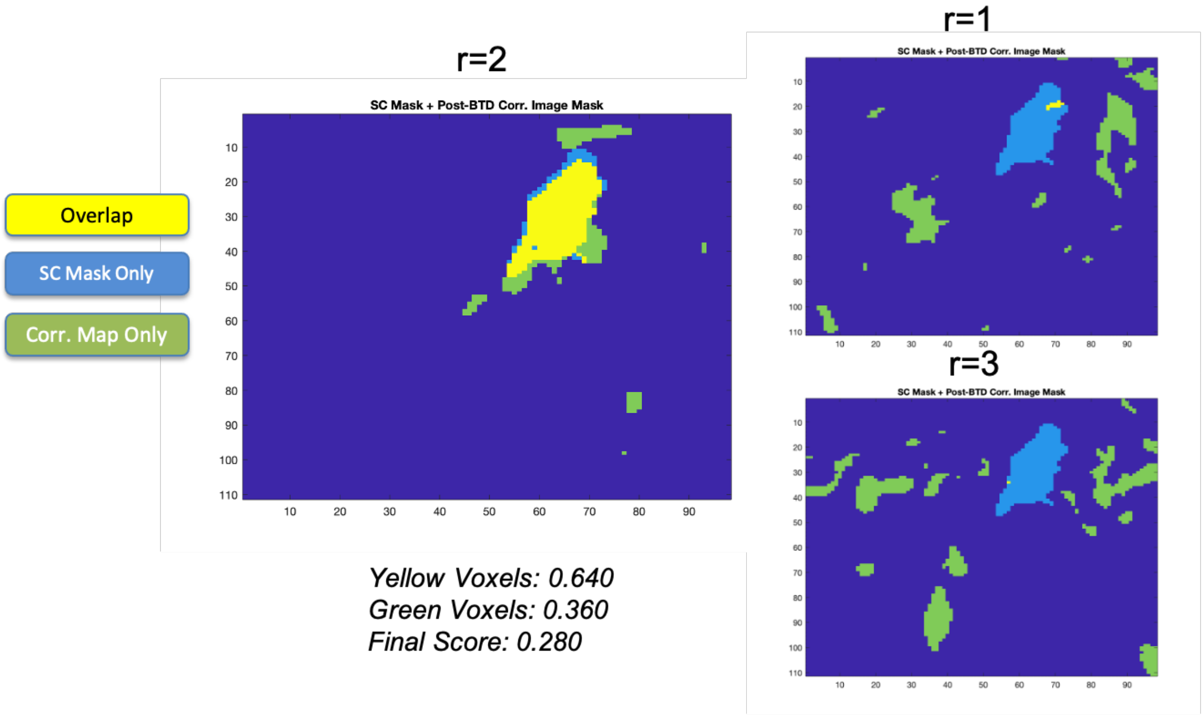


Figure 4.18. Best run results picked out of 100 runs of BTD on the full mixture data. All three correlation maps generated from extracted components are shown here as their binary masks created by using the threshold of 2 times standard deviation. They are overlaid on SC binary mask and the overlapping pixels are shown as yellow. Pixels that remain outside the SC mask is shown as green, while pixels belonging to the SC mask that remain outside of the correlation map mask is shown as light blue. A final score is computed by subtracting number of green pixels from number of yellow pixels, divided by total number of pixels of the correlation map. The best run is picked by looking at this score. Only for one component significant overlap with SC mask is seen, while the others show minimal overlap. The component that shows maximum overlap is regarded as the component that carries information common for both slices.

It can be observed that the component that we know to be active in correlation with the stimulus in both slices, the SC is successfully extracted in one component. This component can thus be regarded as $\boldsymbol{S}_{common}$ in Equation 4.10 while the others can be regarded as carrying uncommon information that belong to each slice.

However, it is unclear that this successful extraction of SC is indeed due to the fact that we are incorporating multiple slices or it is due to the fact that we're simply using more frames from two slices. That is, if these results can be improved by taking a longer recording as well instead of incorporating frames from different slices. In order to investigate this, we first look at the decomposition made from using only the first half of the data, and using only the first slice.

Figure 4.19 shows the result of decomposing only the first half of the single slice data, taking only the first slice. On the left, thresholded correlation map of the component that gave the best score is compared against the binary mask, showing minimal overlap. On the right, histogram produced by accumulating best score that came out of each of 100 trial runs are given. Each data point in these histograms correspond to the score of the component that gave the best score for each trial run, which is assumed to be corresponding to $\boldsymbol{S}_{common}$.

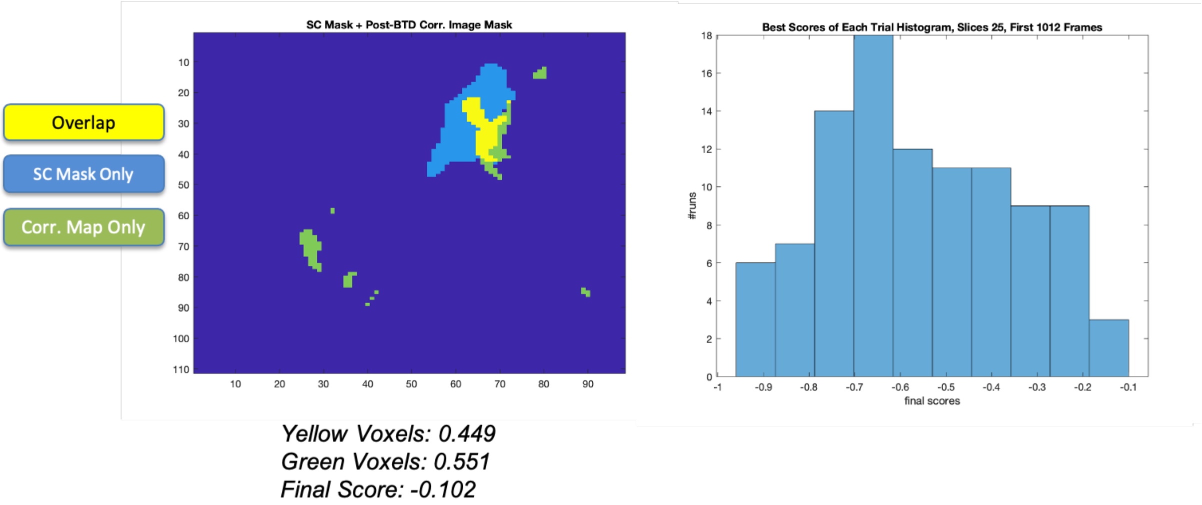


Figure 4.19. Results obtained by only processing slice 25, first 1012 frames. Left) Binary mask obtained from correlation map overlaid on binary mask of the SC region. The run that gave the best score out of 100 runs of BTD is given. Right) Histogram of scores belonging to the component that gave the best score in each run is given.

Figure 4.19 shows the result of decomposing only the first half of the single slice data, taking only the second slice.

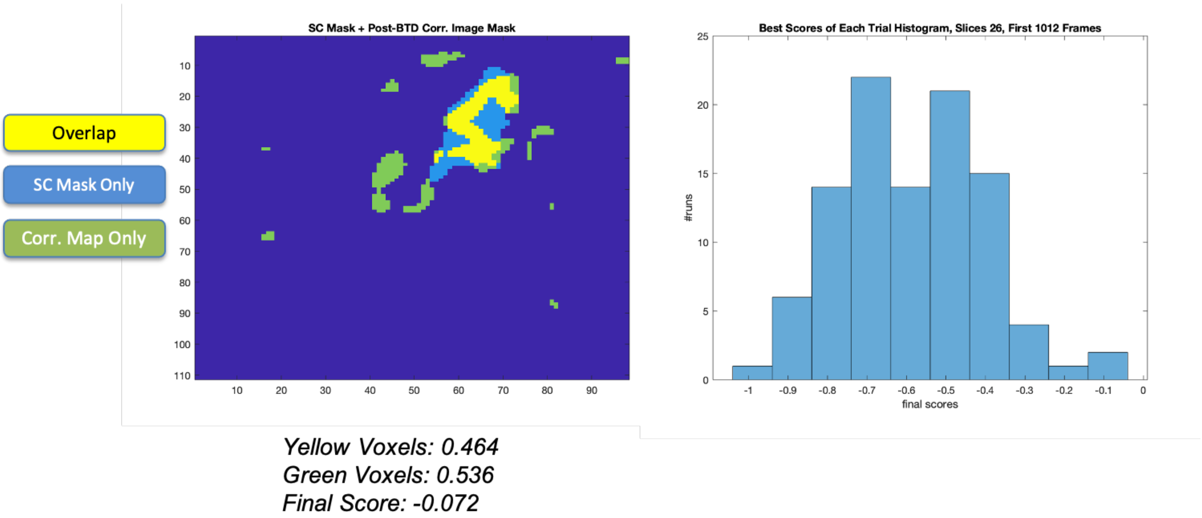
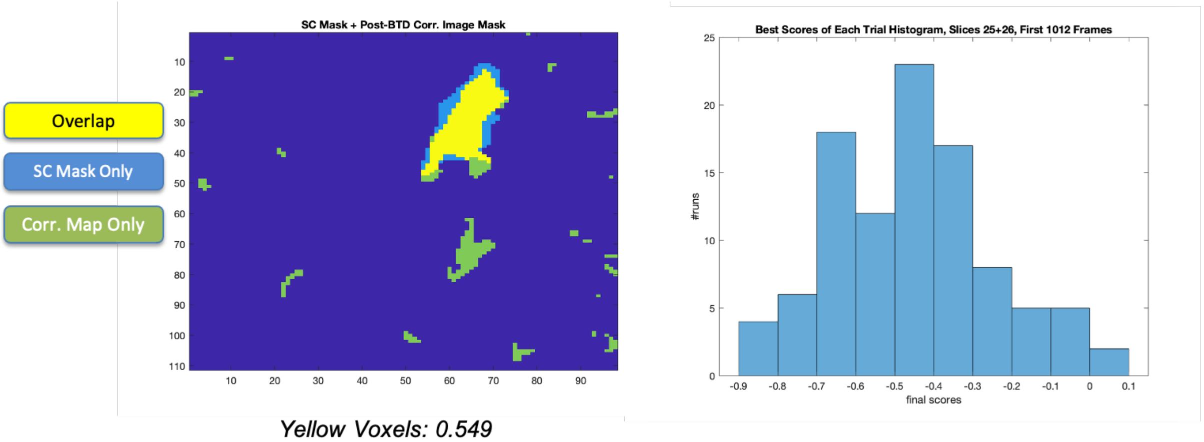


Figure 4.20. Results obtained by only processing slice 26, first 1012 frames. Left) Binary mask obtained from correlation map overlaid on binary mask of the SC region. The run that gave the best score out of 100 runs of BTD is given. Right) Histogram of scores belonging to the component that gave the best score in each run is given.

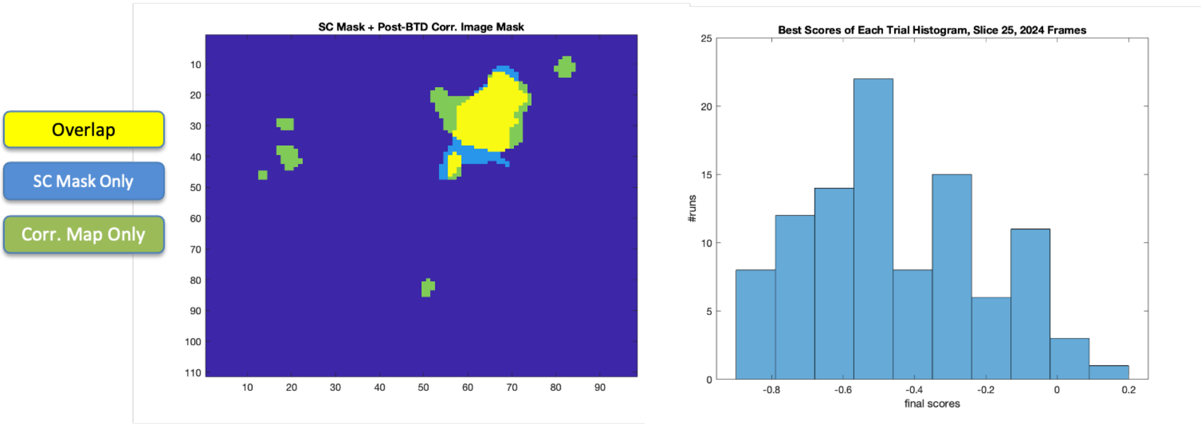
Figure 4.21 shows the result of decomposing only the first half of the data formed by summing two slices.



Yellow Voxels: 0.549
Green Voxels: 0.450
Final Score: 0.099

Figure 4.21. Results obtained by only processing summation of slice 25 and slice 26, first 1012 frames. Left) Binary mask obtained from correlation map overlaid on binary mask of the SC region. The run that gave the best score out of 100 runs of BTD is given. Right) Histogram of scores belonging to the component that gave the best score in each run is given.

Finally, Figure 4.22 shows the result of decomposing all of the 2024 frames that belong to the first slice.



Yellow Voxels: 0.590
Green Voxels: 0.410
Final Score: 0.180

Figure 4.22. Results obtained by only processing slice 25, all 2024 frames. Left) Binary mask obtained from correlation map overlaid on binary mask of the SC region. The run that gave the best score out of 100 runs of BTD is given. Right) Histogram of scores belonging to the component that gave the best score in each run is given.

A more intuitive way to assess the advantage of incorporating more frames or more slices is to look at the histograms produced for same axes, allowing us to compare them in Figure 4.23

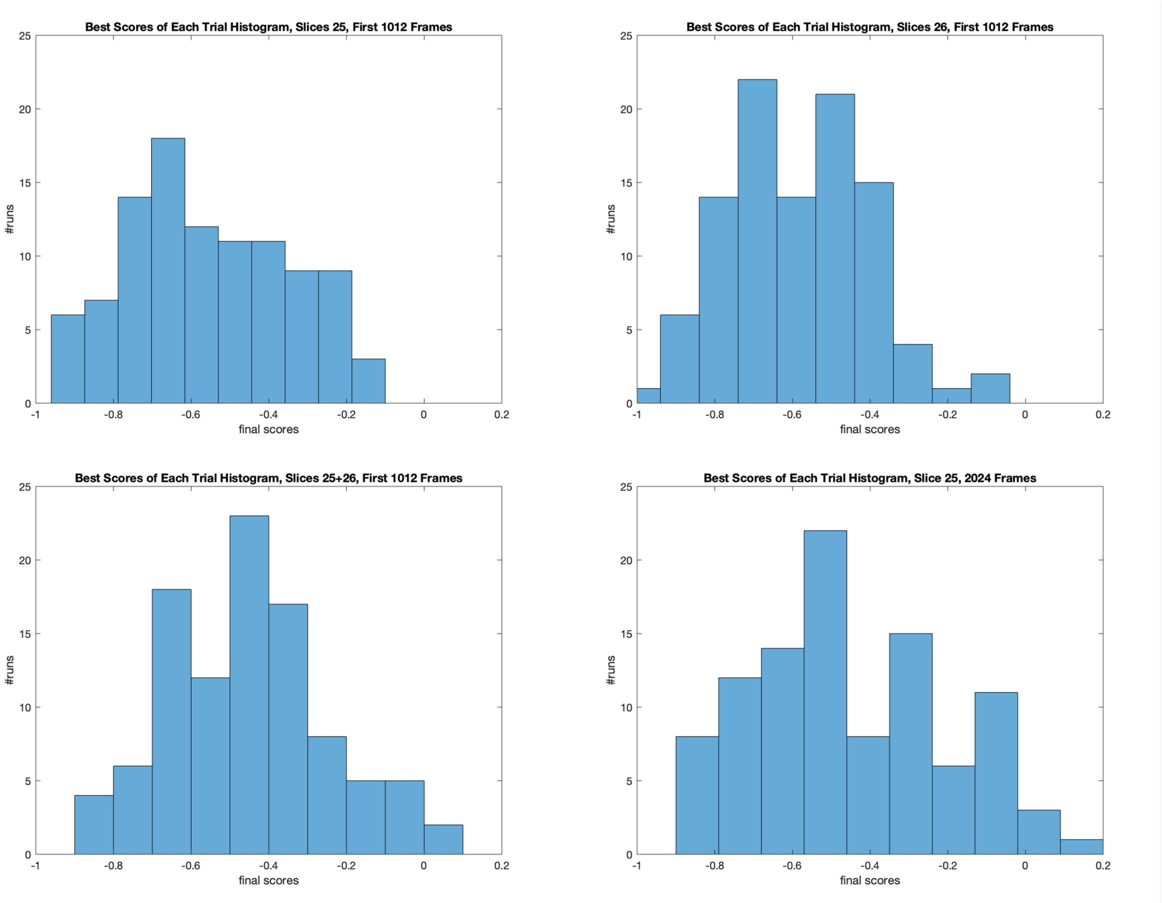


Figure 4.23. Comparison of histograms consisting of scores of components that gave the best score in each run. Top left) Only using slice 25, first 1012 frames. Top right) Only using slice 26, first 1012 frames. Bottom left) Using summation of slices, first 1012 frames. Bottom right) Only using slice 25, all 2024 frames.

Here, we can clearly see that the histograms shift to the right with addition of the other slice as shown in bottom left, or taking a longer recording as shown in bottom right, both improving the best scores obtained for each trial. Thus we can conclude that taking a longer recording produces as good results as incorporating frames from the adjacent slice.

It should be noted that these results are expected to be dependent to the chosen ranks of decomposition, the L and M parameters as well as the thresholding applied to the correlation maps. In this case two times the standard deviation of the images are used as binary thresholds before evaluation.

By decreasing the values of L and M to 10 and 3 we observe that a higher number of lower scores are obtained, indicating that these ranks are statistically less sufficient for represent the active common information regarding the region of SC. By increasing the values of L and M to 30 and 20 we observe that the histogram of best scores shift to the right, giving us better scores. This indicates that a decomposition with given ranks is able to represent the region of SC better compared to the default values used as $R = 20$ and $L = 10$. Histograms produced by setting ranks to different values are shown in Figure 4.24.

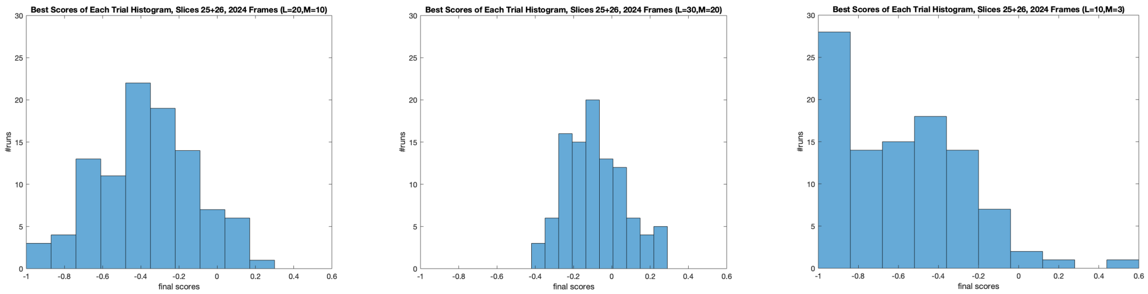


Figure 4.24. Histograms consisting of scores that belong to components that gave the best score in each run is given with the usage of different ranks in L,L,M BTD's. Left) Histogram obtained when $L = 20$, $M = 10$. Middle) Histogram obtained when $L = 30$, $M = 20$. Right) Histogram obtained when $L = 10$, $M = 3$. It is observed that increasing the ranks led to improvement of results whereas decreasing the ranks led to an increase in worse scores closer to -1.

4.8. Discussion

In this chapter, several methods are applied to the fUS data for blind source separation of active regions in the brain, allowing us to see how multi-slice information incorporated into the functional analysis affected the results. It is observed that for ICA, concatenation of pixels from the two slices resulted in the appearance of SC on each side of acquired component, indicating commonality in two slices. However the algorithm failed to generate a stable component with a highly correlated time course from this region, indicating that temporal differences between recordings of the slices are still significant enough to prevent a common highly correlated time course from appearing.

As a method that prevents the potentially unpreferable vectorization, Block-Term Decomposition is introduced and first applied in a simulation case involving two stock photographs, illustrating its capability for blind source separation. It is observed that diverging from the true rank of images or adding noise still leads to successful separation of the photographs, emphasizing the robustness of the algorithm.

After the simulation study, BTD is applied to fUS data by forming a 4-D data structure by stacking the brain slices to a new dimension. Our results demonstrate that avoiding vectorization with BTD did not lead to discovery of new regions, although it illustrated how the components corresponding to active regions can be expressed in a low-rank fashion. However, with chosen ranks and settings, BTD failed to extract all components found by single slice ICA from the 4-D data. One improvement observed here in comparison to single slice ICA results is the higher correlation value produced by the motor cortex, indicating that this region might as well be regarded as a task-induced component.

Lastly, BTD is applied to 3-D data formed by summing the data from two slices, with the goal of amplifying the common information between the slices before decomposition. In this case, evaluation using a binary mask of SC region obtained from ICA results indicate that both taking a longer recording of the same slice or incorporating information from the second slice help to increase the quality of extraction of common information, which is assumed to be activation in SC region here.

It is important to note that all results presented here are dependent to the ranks and parameters that are chosen, and thus different choice of ranks can lead to different results that are not available in this work. Although in 4-D decompositions, we observed similar results when changing R and L values. Both MC and CR was observed in these trials with different ranks similarly in the case of default ranks whose results presented here. However, higher temporal correlation with MC was achieved with only setting $R = 10$ and $L = 50$ other than the default $R = 10$ and $L = 30$. In mixture decomposition case, lower L and M values lead to a higher number of lower scores, leading us to infer that the ranks that are below the default settings fail to represent the correlated region of SC properly. When L is set to a higher value of 30 and M is set to a higher value of 20, the histogram shifts to the right with an increase in better scores, leading us to believe that these ranks estimate the true rank more sufficiently compared to the default values used as $L = 20$ and $M = 10$.

In conclusion, for the chosen ranks and parameters, we fail to show a clear advantage of simultaneous processing of multiple slices as opposed to single slice processing with the exception of higher correlation value captured from MC in 4-D BTD trials. The ineffectiveness of the results of multi-slice analysis compared to single-slice is also apparent in the case of ICA. In comparison to non-vectorized approaches using BTD, single slice ICA seems to both extract more anatomically reasonable components that are consistent between slices, and achieve higher correlations for the time course of the task induced component of SC.

5

Conclusion

In this work, tensor decompositions as ways of compressing and decomposing fUS data are explored, while addressing two goals. The first was to compress fUS data efficiently, reducing the data size while preserving valuable information, i.e., functional response of the brain related to the stimulus. The second goal was motivated by the limitation of having 2-D sensors to capture data related to an inherently 3-D structure, prompting us to investigate the advantage of incorporating 3-D information via processing multiple slices simultaneously as opposed to separately.

In Chapter 3, we applied sequentially truncated multilinear singular value decomposition in order to compress the fUS recordings. The idea of using a multilinear low-rank representation to compress stemmed from the observation that the functional response that we aimed to preserve is formed by neighbourhoods of pixels that act cohesively in time within the experiment, allowing for a low-rank representation as opposed to the noise that is distributed across the data. This compression method was compared against ensemble averaging used in the conventional pipeline, and was observed to lead to a better compression rate, preserving more temporal resolution for the chosen ranks.

In Chapter 4, blind-source separation methods were applied to single slice and two-slice data, in order to investigate the advantage of multi-slice processing that incorporates 3-D information for extracting spatial maps that correspond to active regions in the brain. First, single slice ICA that uses an independence constraint amongst the extracted components was applied, which we used a benchmark for comparison. Secondly, ICA was applied to two-slice data formed by concatenation of voxels in space, providing us the first example of multi-slice processing. The region of SC, the only task induced component captured in single slice ICA that is known to be common in two slices, was captured in the same component appearing on the left and right of the spatial map. This confirmed its commonality in space in the two slices, however the low correlation value that its time course gave indicated that the assumption of common time courses might not be valid.

Next, Block Term Decomposition was proposed as a way of processing the data as it is, in its inherent 3-D structure without vectorization, which might cause underestimation of the existing interactions between the folded modes. First, a simulation study is performed using two stock photographs in order to illustrate the blind source separation capability of BTD for the dimensions that we will be using. It was observed that the method was able to separate the photographs even when using a rank value below the true rank, as well as in noisy conditions.

Following the simulation study, Block-Term Decomposition is applied to the 4-D data formed by stacking of bran slices in a new dimension, as a method that does not involve vectorization. However, we saw that this method performed worse than single slice ICA in terms of extracting the active regions. However, it did produce the spatial map of MC region with a relatively higher temporal correlation with the stimulus signal compared to single slice ICA, indicating that this region might also be task-induced.

In order to amplify common information between slices, a new 3-D data structure is then formed by summing data related to two slices. For extraction of this common information, a BTD is then applied to this aggregate data with $R = 3$. The selected value for the number of sources was motivated by allocating one source for common information shared between the two slices, and the remaining two sources for the individual information contributed by each slice. Results from this decomposition were evaluated using a mask of SC obtained from ICA trials, leading us to discover that both taking a longer portion of single slice data and incorporating the second slice helps to achieve better scores.

In conclusion, tensor decompositions here provided a promising way of compressing and decomposing data which are both often included in a functional analysis pipeline. Utilizing the low rank structure of desired signal related to activity of the brain, we were able to represent it using a low rank multilinear decomposition in both cases, either for compression by truncation or decomposing to show different active regions. However, for the data and experiment at hand, we failed to show a conclusive evidence regarding the advantage of simultaneous processing of slices as opposed to single slice processing, despite achieving better correlations for the region of Motor Cortex. Single slice ICA extracted more anatomically reasonable regions that were consistent between slices, including SC as a task-induced region.

5.1. Future Work

In Chapter 3, the ranks that gave the best ratio of compression rate and quality of preservation of functional response were chosen by trial and error, constituting an exploration on how to process the data at hand with this compression scheme. Future work may address the problem of choosing ranks automatically for any kind of data by developing a method based on multilinear singular values obtained from the decomposition.

In Chapter 4, we attempted to illustrate the advantage of processing slices simultaneously over single slice processing, yet no conclusive evidence was found with chosen ranks and parameters. Future work may address the same problem by choosing different parameters (especially for L and R) for BTD to investigate this further.

An experimental setup that might be used for assessing the advantage of multi-slice processing may involve two probes that record each slice simultaneously, alleviating the differences between recordings that stem from recording at different time instances. In such setting, the true potential of processing the data in its 3-D form in space may be uncovered. Although this approach might still prove to be ineffective if differences are inherent to the responses in each slice i.e. the parts of the regions captured in slices have different delays themselves.

On the analysis side, epoch based analysis using time locked segments on the onset of each response peak could reveal commonality in time courses despite different delays. In such analysis it is required to detect the average response peak with high temporal precision. Thus raw beamformed frames denoised by ST-MLSVD compression could be utilized for such task rather than using PDIs. Alternatively, the tensorization obtained by stacking epochs on a new dimension can also be used with different decomposition techniques such as CPD or BTD in order to reveal epoch-based differences of the functional brain response.

Bibliography

[1] B. Hunyadi, P. Dupont, W. Van Paesschen, and S. Van Huffel, “Tensor decompositions and data fusion in epileptic electroencephalography and functional magnetic resonance imaging data,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 7, no. 1, p. e1197, 2017.

[2] C. Chatzichristos, M. Davies, J. Escudero, E. Kofidis, and S. Theodoridis, “Fusion of eeg and fmri via soft coupled tensor decompositions,” in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 56–60.

[3] L. De Lathauwer, B. De Moor, and J. Vandewalle, “On the best rank-1 and rank-(r 1, r 2,..., rn) approximation of higher-order tensors,” *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000.

[4] G. Northoff and Z. Huang, “How do the brain’s time and space mediate consciousness and its different dimensions? temporo-spatial theory of consciousness (ttc),” *Neuroscience & Biobehavioral Reviews*, vol. 80, pp. 630–645, 2017.

[5] S. Rabanser, O. Shchur, and S. Günnemann, “Introduction to tensor decompositions and their applications in machine learning,” *arXiv preprint arXiv:1711.10781*, 2017.

[6] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[7] L. Sorber, M. Van Barel, and L. De Lathauwer, “Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-(l_r,l_r,1) terms, and a new generalization,” *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 695–720, 2013.

[8] Y. Cai and P. Li, “A blind block term decomposition of high order tensors,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, 2021, pp. 6868–6876.

[9] V. D. Calhoun, J. Liu, and T. Adalı, “A review of group ica for fmri data and ica for joint inference of imaging, genetic, and erp data,” *Neuroimage*, vol. 45, no. 1, pp. S163–S172, 2009.

[10] V. D. Calhoun, T. Adali, L. K. Hansen, J. Larsen, and J. J. Pekar, “Ica of functional mri data: an overview,” in *in Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation*. Citeseer, 2003.

[11] E. Mace, G. Montaldo, B.-F. Osmanski, I. Cohen, M. Fink, and M. Tanter, “Functional ultrasound imaging of the brain: theory and basic principles,” *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 60, no. 3, pp. 492–506, 2013.

[12] B. M. Ances, “Coupling of changes in cerebral blood flow with neural activity: what must initially dip must come back up,” *Journal of Cerebral Blood Flow & Metabolism*, vol. 24, no. 1, pp. 1–6, 2004.

[13] S. L. Norman, D. Maresca, V. N. Christopoulos, W. S. Griggs, C. Demene, M. Tanter, M. G. Shapiro, and R. A. Andersen, “Single-trial decoding of movement intentions using functional ultrasound neuroimaging,” *Neuron*, vol. 109, no. 9, pp. 1554–1566, 2021.

[14] C. Demené, T. Deffieux, M. Pernot, B.-F. Osmanski, V. Biran, J.-L. Gennisson, L.-A. Sieu, A. Bergel, S. Franqui, J.-M. Correas *et al.*, “Spatiotemporal clutter filtering of ultrafast ultrasound data highly increases doppler and fultrasound sensitivity,” *IEEE transactions on medical imaging*, vol. 34, no. 11, pp. 2271–2285, 2015.

[15] I. Hashimoto, K. Yoshikawa, and T. Kimura, “Somatosensory evoked potentials to a threshold air-puff can predict stimulus detection in human subjects,” *Neuroscience letters*, vol. 282, no. 3, pp. 181–184, 2000.

[16] S. Ito and D. A. Feldheim, “The mouse superior colliculus: an emerging model for studying circuit formation and function,” *Frontiers in neural circuits*, vol. 12, p. 10, 2018.

[17] M. A. Basso and P. J. May, “Circuits for action and cognition: a view from the superior colliculus,” *Annual Review of Vision Science*, vol. 3, pp. 197–226, 2017.

[18] J. D. Cohen and M. A. Castro-Alamancos, “Behavioral state dependency of neural activity and sensory (whisker) responses in superior colliculus,” *Journal of neurophysiology*, vol. 104, no. 3, pp. 1661–1672, 2010.

[19] Q. Wang, S.-L. Ding, Y. Li, J. Royall, D. Feng, P. Lesnar, N. Graddis, M. Naeemi, B. Facer, A. Ho *et al.*, “The allen mouse brain common coordinate framework: a 3d reference atlas,” *Cell*, vol. 181, no. 4, pp. 936–953, 2020.

[20] H. Andrews and C. Patterson, “Singular value decomposition (svd) image coding,” *IEEE transactions on Communications*, vol. 24, no. 4, pp. 425–432, 1976.

[21] J.-F. Yang and C.-L. Lu, “Combined techniques of singular value decomposition and vector quantization for image coding,” *IEEE Transactions on Image Processing*, vol. 4, no. 8, pp. 1141–1146, 1995.

[22] C. McGoldrick, W. Dowling, and A. Bury, “Image coding using the singular value decomposition and vector quantization,” in *Fifth International Conference on Image Processing and its Applications, 1995.* IET, 1995, pp. 296–300.

[23] P. Waldemar and T. Ramstad, “Image compression using singular value decomposition with bit allocation and scalar quantization,” in *Proceedings of NORSIG Conference*, 1996, pp. 83–86.

[24] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, “Mpca: Multilinear principal component analysis of tensor objects,” *IEEE transactions on Neural Networks*, vol. 19, no. 1, pp. 18–39, 2008.

[25] J. Ye, R. Janardan, and Q. Li, “Gpca: An efficient dimension reduction scheme for image compression and retrieval,” in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 354–363.

[26] Y. Tang, R. Salakhutdinov, and G. Hinton, “Tensor analyzers,” in *International conference on machine learning.* PMLR, 2013, pp. 163–171.

[27] A. H. Phan and A. Cichocki, “Tensor decompositions for feature extraction and classification of high dimensional datasets,” *Nonlinear theory and its applications, IEICE*, vol. 1, no. 1, pp. 37–68, 2010.

[28] N. Vannieuwenhoven, R. Vandebril, and K. Meerbergen, “On the truncated multilinear singular value decomposition,” *Numerical Analysis and Applied Mathematics Section*, 2011.

[29] B. H. Nagaraja, O. Debals, D. M. Sima, U. Himmelreich, L. De Lathauwer, and S. Van Huffel, “Tensor-based method for residual water suppression in $\xrightarrow{\quad\quad}$1 h magnetic resonance spectroscopic imaging,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 2, pp. 584–594, 2018.

[30] M. De Vos, A. Vergult, L. De Lathauwer, W. De Clercq, S. Van Huffel, P. Dupont, A. Palmini, and W. Van Paesschen, “Canonical decomposition of ictal scalp eeg reliably detects the seizure onset zone,” *NeuroImage*, vol. 37, no. 3, pp. 844–854, 2007.

[31] Z. Zhang, G. I. Allen, H. Zhu, and D. Dunson, “Tensor network factorizations: Relationships between brain structural connectomes and traits,” *Neuroimage*, vol. 197, pp. 330–343, 2019.

[32] M. J. McKeown, L. K. Hansen, and T. J. Sejnowsk, “Independent component analysis of functional mri: what is signal and what is noise?” *Current opinion in neurobiology*, vol. 13, no. 5, pp. 620–629, 2003.

[33] A. Hyvärinen and E. Oja, “Independent component analysis: algorithms and applications,” *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[34] M. Enthoven, “Differentiating task-based functional ultrasound signals via data-driven decompositions,” 2021.

[35] V. Calhoun and T. Adali, “Complex ica for fmri analysis: Performance of several approaches,” in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP’03).*, vol. 2. IEEE, 2003, pp. II–717.

[36] F. C. Viola, S. Debener, J. Thorne, and T. R. Schneider, “Using ica for the analysis of multi-channel eeg data,” *Simultaneous EEG and fMRI: Recording, Analysis, and Application: Recording, Analysis, and Application*, pp. 121–133, 2010.

[37] A. Subasi and M. I. Gursoy, “Eeg signal classification using pca, ica, lda and support vector machines,” *Expert systems with applications*, vol. 37, no. 12, pp. 8659–8666, 2010.

[38] E. S. Lein, M. J. Hawrylycz, N. Ao, M. Ayres, A. Bensinger, A. Bernard, A. F. Boe, M. S. Boguski, K. S. Brockway, E. J. Byrnes *et al.*, “Genome-wide atlas of gene expression in the adult mouse brain,” *Nature*, vol. 445, no. 7124, pp. 168–176, 2007.

[39] J. Himberg and A. Hyvarinen, “Icasso: software for investigating the reliability of ica estimates by clustering and visualization,” in *2003 IEEE XIII Workshop on Neural Networks for Signal Processing (IEEE Cat. No. 03TH8718).* IEEE, 2003, pp. 259–268.

[40] M. Bousse, O. Debals, and L. De Lathauwer, “A tensor-based method for large-scale blind source separation using segmentation,” *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 346–358, 2016.

[41] L. De Lathauwer, “Decompositions of a higher-order tensor in block terms—part ii: Definitions and uniqueness,” *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1033–1066, 2008.

[42] N. Vervliet, O. Debals, L. Sorber, M. Van Barel, and L. De Lathauwer, “Tensorlab user guide,” *Available on: http://www. tensorlab. net*, 2016.

[43] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. Phan, “Tensor decompositions for signal processing applications: From two-way to multiway component analysis,” *IEEE signal processing magazine*, vol. 32, no. 2, pp. 145–163, 2015.

[44] 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.

[45] Z. Zeng, T.-Z. Huang, Y. Chen, and X.-L. Zhao, “Nonlocal block term decomposition for hyperspectral image mixed noise removal,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2021.

[46] M. Heindorf, S. Arber, and G. B. Keller, “Mouse motor cortex coordinates the behavioral response to unpredicted sensory feedback,” *Neuron*, vol. 99, no. 5, pp. 1040–1054, 2018.