Machine Learning for Tract Segmentation in dMRI Data

Nguyen Thien Bao
Università di Trento, Italy
tbnguyen@fbk.eu

ABSTRACT

Diffusion MRI (dMRI) data allow to reconstruct the 3D pathways of axons within the white matter of the brain as a set of *streamlines*, called tractography. A streamline is a vectorial representation of thousands of neuronal axons expressing structural connectivity. An important task is to group streamlines belonging to a common anatomical area in the same cluster. This task is known as *tract segmentation task*, and it is extremely helpful for neuro surgery or for diagnosing brain diseases. However, the segmentation process is difficult and time consuming due to the large number of streamlines (about 3×10^5 in a normal brain) and the variability of the brain anatomy among different subjects. In our project, the goal is: first, to design an effective method for tract segmentation task based on *machine learning* and second, to develop an interactive tool to help medical practitioners to perform this task more precisely and easily. We propose a design of the interactive segmentation process, consisting of two steps: tract identification and tract refinement. The tract identification step generates the first hypothesis of segmentation to avoid the expert to start segmenting from the whole tractography. This step uses the manual segmentation examples from experts to create the candidate of tract segmentation, and is conceived as a supervised learning task. The next step aims at refining the proposed segmentation and takes place by removing or adding streamlines. With the goal to aid medical practitioners to perform this refinement task more precisely and easily, it is necessary to cluster some *similar* streamlines into one set, called *bundle*. We design it as a clustering task. Some of our preliminary results are used for clinical usecase, such as finding the difference between healthy and ALS (Amyotrophy Lateral Smytrophic) diseased brains. Based on this, we believe that with our work, the task of tract segmentation can be performed more easily, at an acceptable computational cost, high accuracy, and can bring benefit for clinical applications.

Keywords

Machine Learning, Tract Segmentation, Brain Connectivity, dMRI Data, Neuro Imaging

1. INTRODUCTION

In neuroimaging, brain connectivity refers to building a model of the connections between different brain areas. *Functional connectivity* focuses on the temporal correlation between the brain activity of anatomically remote areas. *Effective connectivity* investigates causal links between different brain structures. *Anatomical connectivity* refers to the structural links between different areas that develop in the white matter of the brain. Anatomical connectivity is the main focus of this work. Currently, diffusion magnetic resonance imaging (dMRI) techniques are commonly used to find the anatomical connectivity in brain [2, 26]. dMRI is a set of methods for measuring the displacement distribution of water molecules in vivo. From the displacement distribution, we can infer the fiber orientation or orientations in each imaging volume element, called *voxel*, and that allows to reconstruct white matter fiber tracts as a set of *streamlines*. A streamline is an approximation of about $\sim 10^3$ neuronal axons sharing the same structural connectivity path. A set of streamlines with similar spatial and shape characteristics is called *bundle*. The whole set of streamlines of a brain is called *tractography*. More recently, several groups have proposed tractography methods and reported success in following fiber tracts (for example deterministic tractography algorithms [18],[7]). One problem raises is how to group streamlines belonging to a common anatomical area into one segmentation. This task is known as *tract segmentation*.

Traditionally the segmentation task is done by neuroanatomists, and it consumes a lot of time and effort due to the large number of streamlines (about 3×10^5 in a normal brain). Moreover, the variability of the brain anatomy among different subjects makes the segmentation become a difficult task [3]. Recently, the literature about machine learning techniques to solve this problem is increasing. Up to now, there are two approaches for tractography segmentation: supervised [5] and unsupervised [11] learning. The unsupervised techniques often rely on expert-crafted streamline-streamline distance functions [6, 32] encoding informative relationships for the segmentation task, then followed by a clustering algorithm (agglomerative, k-means, Gaussian mixture model, etc. see [29] for a recent brief review). Supervised tract segmentation [5, 21] instead aims at learning how to segment the tractography from expert-made examples provided as input. Although both supervised and unsupervised techniques get some encouraging results, but they are below the expectation of medical practitioners. Unsupervised techniques usually work on the whole tractography while medical practitioner often focus on a specific tract. In the case of supervised learning, the lack of ground truth data makes the results be not good and need the refinement from experts.

This research is motivated by the need for interaction of medical pratitioners when they do the tract segmentation task. It is impossible for them to have a look at the whole tractography of a brain due to the huge number of streamlines ($\approx 3 \times 10^5$). And it is often an important requirement to view the tractography at different partial views. Medical practitioners usually changes between these levels for better visualization. This demand raises a question of how we can present the whole tractography at different level of zooming in $3D$ space?

Currently, both supervised and unsupervised learning are unsatisfactory. In this work, we want improve the support of machine learning for the segmentation task. In another way, we want to help medical practitioner to do the segmentation task more easily, and more accurately based on machine learning. More precisely, following are things we want to investigate in this project.

Brain tractography segmentation: In this work, we combine both *unsupervised* and *supervised* learning to do the task of segmentation. The reason is that each technique is suitable for different steps in the whole process. The framework of our process is showed in the figure 2. First, supervised learning is used in the tract identification stage to create the tentative segmentation. Then, this candidate is interactively refined by experts based on fast clustering technique. ***Multimodal brain image visualization:*** In order to help medical practitioners to do the segmentation task more easily, and faster, we provide an interactive visualization tool for a large volume brain imaging data in $3D$ space, which is the implementation of our proposed framework for tract segmentation. In this scientific interactive tool, we present a simple way to interact and to segment streamlines, which goes, as far as we know, beyond any other available medical imaging software. ***Clinical applications:*** The result of tract segmentation is applied to computer aided early diagnosis brain diseases. At a preliminary application, we want to find what differs between the healthy brain and the diseased brain of patients with the ALS disease (Amyotrophic Lateral Sclerosis)[1].

The structure of this paper is as follows. In Section 2, we introduce the basic background of dMRI and tractography segmentation. The latter part of this section briefly summarizes some current trends in segmentation tractography task. Section 3 formally presents the problem of clustering a tractography, which is one of the contribution of this project. In Section 4, we describe the proposed solution for online clustering tractogprahies. In this part, we also illustrate the dissimilarity representation based on the most common streamline-streamline distance functions. In Section 5, we present the preliminary result of our experiment about the dissimilarity approximation on a real dMRI dataset from the University of Cambridge (UK). An interactive visualization software tool for segmentation, called **Spaghetti**, is also presented. Moreover, one clinical application to support the diagnosing of the ALS-disease is also introduced. The last section, Section 6, discusses results and points out some future works.

2. STATE OF THE ART

Recently, several groups have proposed tractography methods for reconstruction the whole brain tractography from dMRI data [32]. The most popular is the **deterministic tractography** algorithms [18] An example of the partial tractography extracting from dMRI by using deterministic algorithm is shown in figure 1. However, the resulting tractography datasets are highly complex and include thousands of fibers (about $\approx 3 \times 10^5$), which requires techniques or method to create the exact anatomic brain before doing further studying. The **segmentation** aims at doing this task, groups some fiber tracts belonging to a common anatomical area into one segmentation. Due to the fact that the complex tractography datasets present a large amount of short association bundles which have been rarely studied until now, it is extremely difficult to cluster some fibers having the same anatomical structure into a group. The segmentation of fiber bundles is therefore a complex and not completely solved problem. Recently, the literature about machine learning techniques to solve this problem is increasing. In the following part, the brief survey about currently trends in segmentation tractography are presented.

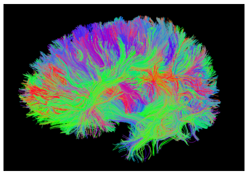

Figure 1: A tractography of $\approx 3 \times 10^5$ streamlines within a brain. Each single streamlines represents thousands of neighboring neuronal axons expressing structural connectivity. Only 3% of the streamlines are shown to improve readability.

Atlas approach Atlas are the models of white mater structure in brain. Firstly, atlas are created from experience of experts without being driven from data. After that, atlas are used as model of clusters for tractography segmentation. All streamlines would be grouped into the closest cluster in atlas. O'Donnell and Westin [20] generated a tractographic atlas using spectral embedding and expert anatomical labeling. They then automatically segmented the new tractography using again spectral clustering and embedding the tracks as points in the embedded space, to the closest existing atlas clusters. The true affinity matrix was too big to compute therefore they used the Nystrom approximation: working on a subset and avoid generating the complete distance matrix. However, the important information from the full data set may be lost after sub-sampling. The other main issue of this approach is that up to present, there is no believable algorithm for co-registration two anatomical brains due to the difference of size, position, and direction of these brains.

ROI - region of interest One of the first idea for segmentation is to use the region of interest (ROI) [28]. This approach tried to reconstruct tracts passing through ROI by exploiting existing anatomical knowledge of tract trajectories. First, some target tracts must be defined. It also requires to specified manually some regions where tracts start, end or pass through. Then streamlines would be filtered based on the constraint of passing through ROIs. ROI approach needs a priori knowledge about the trajectory and is used only for well-characterized white matter tracts. In order to refine the segmentation, multi-ROIs were used to include or exclude tracks.

Unsupervised learning From the point of view of algorithmic approaches, the segmentation task has traditionally been addressed with unsupervised techniques over only diffusion data [32]. This typical framework first defines a pairwise distance between fibers and inputs the similarity matrix to standard clustering algorithms. Various distance functions between fibers have been

[1] http://www.alsa.org/about-als/what-is-als.html

proposed: the Euclidean distances between fiber shape descriptors [4]; the similarity between two fibers based on the number of points sharing the same voxel [13]; distance from the B-spline representation [15] ; closest point distance, mean of closest distances and Hausdorff distance [10]. Then, following is a clustering algorithm (agglomerative, k-means, Gaussian mixture model, etc. see [29] for a recent brief review of applying these algorithm for tractography).

The disadvantage of these clustering algorithms is that they require manually specifying the number of clusters or a threshold for decide when to stop merging or splitting clusters. The different numbers of chosen clusters vary significantly the performance of clustering [17]. Recently, there are some approaches try to solve this problem by auto choosing the number of clusters. In [20], a large cluster number for spectral clustering is chosen, and then these clusters are manually merged to obtain models for white matter structures. Zvitia et al. [33] and Wassermannet al. [30] decide the number of clusters based on mean-shift. By adding a penalty to a larger cluster number, Neji et al. [19] solved the optimization using linear programming to chose the number of clusters. Recently, Garyfallidis et al. [8] proposed a very quick clustering algorithm, called QuickBundles. It took one random streamlines as initial cluster, and calculated the distance from all the un-clustered streamlines to the representatives of clusters. Only the streamline with the minimum distance was grouped into the closest cluster if the distance was less than a given threshold, other while, that streamline became a new cluster.

Although these approaches avoid manually choosing number of cluster, the drawback is the high space and time complexities of computing pairwise distances between fibers. Whole brain tractography produces $\approx 3 \times 10^5$ streamlines fibers per subject, it is difficult to compute, and it becomes more serious when clustering fibers of multiple subjects. To avoid computing pairwise distances between fibers, Savadjiev et. al. [25] clustered diffusion orientation distribution functions maxima instead of clustering fiber tracts directly. This algorithm based on the geometric coherence of fiber orientations. Maddah et al. in [16] proposed a probabilistic approach to cluster fibers. It used a Dirichlet distribution as a prior to incorporate anatomical information. However, this algorithm also required establishing point correspondence which was difficult to define.

The most disadvantage of unsupervised approach is that it works on the whole tractogrpahy and tries to cluster tractography into many tracts. While the requirement of medical practitioners only focuses on some specific tracts.

Supervised learning Supervised segmentation is the method of partitioning according to provided examples. Firstly, the target tracts should be specific, such as corticol spinal tracts (see figure 6. Then, a repository of samples must be collected. A sample is an expert-made assignment of streamlines to the target tracts. These samples are used to train a classify model, which is used to cluster a new streamline. In this setting, each streamline can be class-labelled as being member of the fiber tract of interest or not. For this reason the supervised segmentation problem becomes a binary classification problem.

Up to present, there is a little attention in the literature about supervised tract segmentation. Maddah et al. [15] used the B-spline representation of the streamlines, and classified by the nearest-neighbor algorithm with respect to an atlas. In [20], O'donnel created an atlas from training dataset based on spectral clustering. Wang et al. [29] proposed a non-parametric Bayesian framework using a hierarchical Dirichlet processes mixture (HDPM) model. The models of bundles were learned from how voxels are connected by fibers in training data instead of comparing fiber distances. Olivetti [23] combined both structural and functional connectivity to study jointly in a pairwise approach with the goal of assessing the contributions of structural information and functional information when segmenting the tracts. Recently, [21] solved this classification problem basing on the dissimilarity representation. After projecting all streamlines into some prototypes, one streamline-streamline distance function is computed in this new representation space, and it is used for classifying.

Although supervised approaches focus on a specific tracts as requirement of medical practitioners. However, because the number of data for training and testing is very small due to the vague time for collecting enough the truth background data of manual segmentation tractography, the results usually are bellow the expectation of medical practitioners, and they need to be refined to use in clinical applications.

The drawback of these approaches is that or they work on a large number of tracks and most of them are not interested to medical practitioners; or they focus on a target tract but the variance between brains makes it difficult to generate well. The results from both case are needed the refinement from expert. In this work, we want to combine both supervised and unsupervised approach to overcome these disadvantage. Moreover, we propose a framework using **BOI**(Bundles of Interest). While ROI concerns about which streamlines go through some interesting regions, BOI focuses only on streamlines inside some specific bundles. Because all of the current approaches only work on the tracks without caring the anatomy, and it makes difficult to validate the result. Using BOI would make medical practitioners concentrate on which tracts they are working on, and of course these tracts also correlate to the anatomy. Moreover, while all the current methods are off-line and medical practitioners can not interact or modify the result of segmentation, in this project, we want to build a tool that can help them instantly to refine the segmentation result mannualy. It is also another novelty of our approach. Beside, this tool should has ability of real time adapting to the responding of users. This is also the other different of our method to most of the state-of-the-art approaches, which can not adjust to the user feedback. In another way, in this project, our goals are to:

- First, design an effective method for tract segmentation task using *machine learning* based on BOI approach.
- Second, develop a scientific interactive visualization tool, which is the implementation of the framework that we propose for tract segmentation task in the previous step.

With the assistance of this scientific interactive tool, we have a strong belief that the task of tract segmentation would be done more precisely, more easily and would consume less time.

3. PROBLEM STATEMENT

Basic notation: Let the polyline $s = \{\vec{x_1}, \dots, \vec{x}_{n_s}\}$, where $\vec{x} \in \mathbb{R}^3$, be a *streamline* reconstructed from dMRI data by deterministic tractography algorithms [18]. Let the *tractography* $\mathbb{T} = \{s_1, \dots, s_n\}$ be defined as a set of n streamlines. Let τ be an anatomical fiber tract of interest, e.g. the cortical spinal tract (see figure 6), and let $\mathsf{T} \subset \mathbb{T}$ be its corresponding streamline-based approximation within given the tractography.

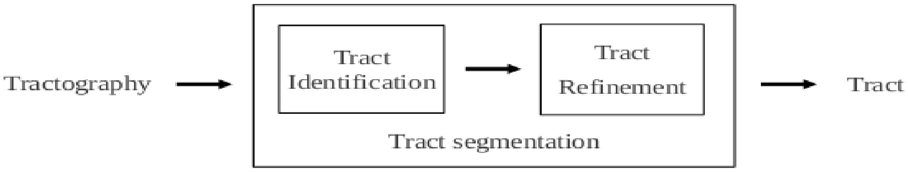

Figure 2: A process design of segmentation task, including two steps: tract identification and tract refinement. The tract identification creates the candidate of segmentation from a repository of examples using a supervised algorithm. The candidate then will be refined by experts with the help of the fast clustering technique.

Process design: we propose a design of interactive segmentation process based on two steps(figure 2): tract identification based on supervised learning and tract refinement based on unsupervised learning. The *tract identification step* generates the first hypothesis of segmentation instead of starting from the whole tractography. It uses the manual segmentation examples from experts to create the candidate of tract segmentation. The tract identification step corresponds to a mapping $f : \mathbb{T} \mapsto \{0, 1\}$ where

$$f(s) = \begin{cases} 1 & \text{if } s \text{ in } \mathsf{T} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

In the machine learning terminology, the function f is called *classifier* and each pair $(s, f(s))$ is a class-labeled *example*. In practice f is not available and the problem is then to infer an approximation g from data. We may have many samples of T for the same fiber tract τ, e.g. the cortical spinal tract, when the annotation is operated by different neuroanatomists. The labelling is prone to error and has to be considered an approximation of the true fiber tract. A classifier g is learned by training a classification algorithm from the set T. Detail about supervised classifier for tractography can be found in [21]. Our main focus in this project is not on this step, but on the next step.

Let $\mathcal{T} \subset \mathbb{T}$ and $\mathcal{T} = \{\text{s} \mid g(s) = 1, \forall s \in \mathbb{T}\}$. In an ideal case, $\mathcal{T} \equiv \mathsf{T}$, but it rarely happens due to the error of function $g(s)$ during the training stage. The *tract refinement step* aims at refining the $\mathcal{T}$ for being close to the T by excluding any unnecessary streamlines or selecting additional ones. In contrast to the previous step, which automatically done by learning from a repository of examples, this refinement step is manually performed by medical practitioners. When they do the segmentation, it is often an important requirement of viewing $\mathcal{T}$ at different levels of grouping for better visualization in detail. This demand raises a question of how to present $\mathcal{T}$ in different level of abstraction. This requirement is much stricter than randomly sampling representatives of $\mathcal{T}$ and hiding the others. In another way, we need a clustering algorithm which has a capability of producing different partitions of $\mathcal{T}$ corresponding to different levels of viewing. We consider it as an unsupervised learning problem.

Data representation: However, most of the state-of-the-art learning techniques (both supervised and unsupervised) often require the data to lie in a vectorial space, which is not the case of streamlines. Streamlines are polylines in 3D space. Each streamline $s = \{\vec{x_1}, \ldots, \vec{x}_{n_s}\}$, where $\vec{x} \in \mathbb{R}^3$, has different length and different number of points, and for this reason they cannot be directly represented in a common vectorial. The lack of the vectorial representation avoids the use of some of these algorithms and of computationally efficient implementations. In this case, we need to find a representation ϕ of streamline in a vectorial space $\phi : \mathbb{T} \mapsto \mathbb{R}^d$, where d is the dimension of the new space. This representation ϕ maps a streamline s from its original space $\mathbb{T}$ to a vector of $\mathbb{R}^d$. This representation is a *lossy* one in the sense that in general it is not possible to exactly reconstruct s from $\phi(s)$ because some information is lost during the projection. How we can minimize this *lossy* is really a challenge. Beside, due to the huge number streamlines, the number of partitions and clusters are also very large. It makes storing and accessing partitions and clusters hard and resource consuming. Therefore, an efficiency structure for representation, storing and accessing clusters and partitions is another important requirement which needs to be fulfilled.
After projecting streamlines into a vectorial space $\mathbb{R}^d$ by using an efficient representation ϕ, the next step is how we can cluster streamlines (in representation space $\phi(s)$) into different clusters corresponding to different levels of viewing.In the following part, we will first present the formal definition of clustering problem, and then state the problem of tractography clustering.

Clustering problem: Clustering is a division of data (or data representation) into a certain number of clusters (groups, subsets, or categories). Up to present, the definition of clustering has still not been agreed universally. Most researchers describe a cluster by considering the internal homogeneity and the external separation, i.e., the similarity between objects within a subgroup is larger than the similarity between objects belonging to different subgroups. Both the similarity and the dissimilarity should be defined in a clear and meaningful way. Here, we give some simple mathematical descriptions of several types of clustering, based on the description in [31].
Given a set of input patterns denoted as $\mathcal{X} = \{\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{j}, \ldots, \mathsf{x}_\mathsf{N}\}$ where $\mathsf{x}_\mathsf{j} = (x_{j1}, x_{j2}, \ldots, x_{jd})^T \in \Re^d$ and each measure x_{ji} is said to be a feature (attribute, dimension, or variable). Clustering attempts to seek a K-partition of $\mathcal{X}$, $C = \{C_1, \ldots, C_K\}$, with $K \leq N$, such that

$$C_i \neq \varnothing, i = 1,\ \ldots,\ K \text{ and } \cup_{i=1}^{K} C_i = \mathcal{X} \tag{2}$$

The difference of the definition of the separation between clusters leads to different types of clustering. *Hard partitional clustering* accepts no overlaping between partitions. All clusters are exclusive, so that each patterns only belongs to one cluster $C_i \cap C_j = \varnothing$, $i, j = 1,\ \ldots,\ K$, and $i \neq j$. *Hierarchical clustering* tries to construct a tree-like nested structure partition of $\mathcal{X}$: $H = \{H_1, \ldots, H_Q\}$, with $Q \leq N$, such that $C_i \in H_m, C_j \in H_l, m > l$ imply $C_i \in C_j$ or $C_i \cap C_j = \varnothing$, for all $i, j \neq i, m, l = 1, \ldots, Q$. *Fuzzy clustering* allows one pattern to belong to all clusters with a degree of membership, $\mu_{i,j} \in [0, 1]$, which represents the membership coefficient of the jth object in the ith cluster and satisfies the following two constraints: $\sum_{i=1}^{K} \mu_{i,j} = 1, \forall j$ and $\sum_{j=1}^{N} \mu_{i,j} \leq N, \forall i$

Distance and similarity: It is natural to ask what kind of standards we should use to determine the closeness, or how to measure the distance (dissimilarity) or similarity between a pair of objects, an object and a cluster, or a pair of clusters. A data object is described by a set of features, usually represented as a multidimensional vector.

A distance or dissimilarity function on a data set $\mathcal{X}$ between individuals is defined to satisfy the following conditions.

$$(d(\mathsf{x_i},\mathsf{x_j}) = d(\mathsf{x_j},\mathsf{x_i})) \wedge (d(\mathsf{x_i},\mathsf{x_j}) \geq 0), \forall \mathsf{x_j}, \mathsf{x_i} \tag{3}$$

if two following conditions (triangle inequality 4 and reflexity 5) are still hold, the distance is called a metric

$$d(\mathsf{x_i},\mathsf{x_k}) + d(\mathsf{x_k},\mathsf{x_j}) \geq d(\mathsf{x_i},\mathsf{x_j}), \forall \mathsf{x_i}, \forall \mathsf{x_j}, \forall \mathsf{x_k} \tag{4}$$

$$d(\mathsf{x_i},\mathsf{x_j}) = 0 \leftrightarrow \mathsf{x_j} \equiv \mathsf{x_i} \tag{5}$$

Likewise, a similarity function is defined to satisfy the conditions in the following.

$$(s(\mathsf{x_i},\mathsf{x_j}) = s(\mathsf{x_j},\mathsf{x_i})) \wedge (0 \leq s(\mathsf{x_i},\mathsf{x_j}) \leq 1), \forall \mathsf{x_j}, \mathsf{x_i} \tag{6}$$

if two following conditions are still satisfied, it is called a similarity metric

$$[s(\mathsf{x_i},\mathsf{x_k}) + s(\mathsf{x_k},\mathsf{x_j})]s(\mathsf{x_i},\mathsf{x_j}) \geq s(\mathsf{x_i},\mathsf{x_k})s(\mathsf{x_k},\mathsf{x_j}), \forall \mathsf{x_i}, \mathsf{x_j}, \mathsf{x_k} \tag{7}$$

$$s(\mathsf{x_i},\mathsf{x_j}) = 1 \leftrightarrow \mathsf{x_j} \equiv \mathsf{x_i} \tag{8}$$

Based on the definition (dis)similarity between two objects, there are many way to define the (dis)similarity between a object and a cluster, or a pair of clusters. A popular group of distances is the modified Hausdorff distances [6]. See [32] for a recent survey about these distances.

For a data set with N input patterns, we can define an $N \times N$ symmetric matrix, called proximity matrix, whose (i, j)th element represents the (dis)similarity measure for the ith and jth patterns $(i, j = 1, \ldots, N)$. Obviously, the (dis)similarity measure directly affects the formation of the resulting clusters. Almost all clustering algorithms are explicitly or implicitly connected to some definition of proximity measure. Some algorithms even work directly on the proximity matrix. Once a proximity measure is chosen, the construction of a clustering criterion function makes the partition of clusters an optimization problem, which is well defined mathematically, and has rich solutions in the literature. The survey of clustering algorithms and distance functions can be found in [31, 24]. Different approaches usually lead to different clusters; and even for the same algorithm, parameter identification or the presentation order of input patterns may affect the final results. Therefore, it is important to carefully investigate the characteristics of the problem at hand, in order to select or design an appropriate clustering strategy.

Clustering tractography Given a set of N streamlines $\mathcal{T} = \{s_1, \ldots, s_N\}$ [2], traditional approaches for clustering tractography usually find a partition $C = \{C_1, \ldots, C_K\}$ with $K \leq N$, satisfying two conditions of clustering 2 $C_i \neq \varnothing, i = 1, \ldots, K$ and $\cup_{i=1}^{K} C_i = \mathcal{T}$ (see more in section 2). However, in this work, as declared before, we want to seek not only one partition but instead, a set m partitions of a $\mathcal{T}$: $\mathbb{P} = \{P_1, P_2, \ldots, P_m\}$, where $P_i = \{C_1^i, C_2^i, \ldots, C_{d_i}^i\}$ is one partition of $\mathcal{T}$ (d_i is the number of clusters in partition P_i). Each partition P_i represents for the ith level of abstraction of $\mathcal{T}$. Obviously, within one partition P_i, there is no intersection between two clusters: $C_k^i \cap C_l^i = \varnothing$, with $\forall k, i \in [1, \ldots, d_i], k \neq i$. But it is allowed to overlap between one cluster belonging to one partition with another cluster of the other partition:

$$(C_k^i \cap C_l^j = \varnothing) \vee (C_k^i \cap C_l^j \neq \varnothing), \forall i, j \in [1, .., m], i \neq j \tag{9}$$

The simplest way to solve this problem is to run m times one current clustering algorithm. But take in to account that the number of streamlines is really huge, and most of clustering algorithms often need to calculate pairwise distances of size $N \times N$ where N is the number of tracks. This amount of comparisons puts a massive load on clustering algorithms forcing them to be inefficient and therefore impractical for our purpose. Beside, partitions of $\mathbb{P}$ are not unrelated to each other at all. Supposed that the abstraction of the ith level is higher than jth level. Let P_i and P_j represent for the level of abstraction ith and jth respectively. Then the relationship between clusters $C_l^i \in P_i$ and $C_k^j \in P_j$ can be presented as:

$$\forall P_i, P_j, i \geq j \mapsto [(C_k^j \subset C_l^i) \vee (C_k^j \cap C_l^i = \emptyset)] \tag{10}$$

with $l \in [1, \ldots, d_i], k \in [1, \ldots, d_j]$. The most challenge is to design a clustering algorithm, based on a specific distance function, which is able to create a set of m-partition $\mathbb{P}$ of $\mathcal{T}$, which satisfies condition 10. It makes our method different to most of the state-of-the-art approaches [28, 20], which can produce only one partition of $\mathcal{T}$.

Another raising problem is that, when medical practitioners do the segmentation, eventhough they do select some clusters, they also want to *check* that some neighbor streamlines *"close"* to these clusters should be included into or excluded from the result or not (called *neighbor checking* problem). Let $P_i = \{C_1^i, C_2^i, \ldots, C_{d_i}^i\}$ be the current viewing partition of $\mathcal{T}$, and $\mathcal{T}_s$ be the set of m-selected streamlines $\mathcal{T}_s = \{s_1^s, \ldots, s_m^s\}$. At the beginning, $\mathcal{T}_s = \bigcup_{j=1}^{d_i} C_j^i = \mathcal{T}$. Given a distance threshold θ, let $neighbor(s, \theta)$ be a set of close streamlines of streamline s.

$$neighbor(s, \theta) = \{s_i \mid d(s, s_i) \leq \theta \ \wedge s_i \neq s, \ \forall s_i \in \mathbb{T}\} \tag{11}$$

where $d(s_i, s_j)$ is a distance function between s_i, s_j; and $\mathbb{T}$ is the whole brain tractography. With a streamline set S, neighbor of S is defined as $neighbor(S, \theta) = \bigcup_{s_i \in S} neighbor(s_i, \theta)$.

Removing streamline: Let streamline $s_{rm} \in C_j^i$ be removed from $\mathcal{T}_s$. In this case, only cluster C_j^i is affected and there is no significant change on $\mathbb{P}$:

$$C_j^k = C_j^k \setminus \{s_{rm}\}, \text{ if } C_j^k = \emptyset \text{ then } P_k = P_k \setminus C_j^i, \forall k \in [i, .., m] \tag{12}$$

Adding streamline: Supposed streamline $s_{add} \in neighbor(\mathcal{T}_s, \theta)$ is additionally selected. The new selected set is $\mathcal{T}_s^{'} = \mathcal{T}_s \cup \{s_{add}\}$ and new partition of $\mathcal{T}_s^{'}$ be $P_i^{'}$. Let $\mathbb{P}^{'}$ be the new m-partition set of $\mathcal{T}_s^{'}$: $\mathbb{P}^{'} = \{P_1^{'}, P_2^{'}, \ldots, P_m^{'}\}$, where $P_i^{'}$ is one partition of $\mathcal{T}_s^{'}$ at the *ith* level. We consider how to generate $\mathbb{P}^{'}$ from the current $\mathbb{P}$. Let a triple $< \mathsf{P}, \mathsf{T}, \gamma >$ denote for the partition set P generated from a set of streamlines T according to some constraints in γ. It is clear that $\mathbb{P}$ is generated on $\mathcal{T}$ only based on the constrain γ_1 of equation 10: $< \mathbb{P}, \mathcal{T}, \gamma_1 >$.

Let $d(s, C)$ be the distance between a streamline s and a cluster C. Let $d(C_i, C_j)$ be the distance between two clusters C_i and C_j. Let $d(C_j^i, P_i)$ be the min distance between cluster $C_j^i \in P_i$ with all the other clusters of partition P_i:

$$d(C_j^i, P_i) = min_{k \in [1,..d_i], k \neq j}(d(C_j^i, C_k^i)) \tag{13}$$

[2] note that $s_i, \forall i \in [1, .., N]$ is the representation of the original streamline $s_i^{'}$ through a representation method ϕ: $s_i = \phi(s_i^{'})$

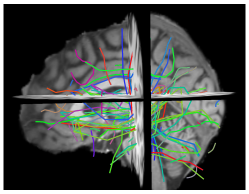

Figure 3: A set of 100 streamlines, i.e. an example of prototypes, from a full tractography

Let $C_{close}(s, P_i)$ be the cluster in P_i closest to streamline s:

$$C_{close}(s, P_i) = argmin_{\forall C_l^i \in P_i}(d(s, C_l^i)) \tag{14}$$

If $d(s_{add}, C_{close}(s_{add}, P_i)) > d(C_{close}(s_{add}, P_i), P_i)$, obviously s_{add} would become a new cluster in partition $P_i^{'}$: $P_i^{'} = P_i \cup \{s_{add}\}$. In this case, $\mathbb{P}^{'}$ is generated from $\mathbb{P}$, with the constraint γ_2: $< \mathbb{P}^{'}, \mathcal{T}_s^{'}, \gamma_2 >$, where γ_2 is the condition of that all the partitions of $\mathbb{P}^{'}$ lower than ith level must be contain $\{s_{add}\}$ as a separative cluster:

$$\forall j \in [1, .., i],\ \exists C'^j_l \in P_j^{'} : C'^j_l = \{s_{add}\} \tag{15}$$

If $d(s_{add}, C_{close}(s_{add}, P_i)) \leq d(C_{close}(s_{add}, P_i), P_i)$, then it is clear that s_{add} would be merged into the cluster $C_{close}(s_{add}, P_i)$: $C_{close}(s_{add}, P_i) = C_{close}(s_{add}, P_i) \cup \{s_{add}\}$. In another word, s_{add} and $C^i_{close}(s_{add})$ must be in the same cluster from the *ith* level of abstraction. For that reason, the set of partitions $\mathbb{P}^{'}$ of $\mathcal{T}_s^{'}$ is driven from $\mathbb{P}$: $< \mathbb{P}^{'}, \mathcal{T}_s^{'}, \gamma_3 >$, with constrain γ_3:

$$\forall k \in [i, .., m], \exists C_j^k \in P_k^{'} : (C_{close}(s_{add}, P_i) \cup \{s_{add}\}) \subseteq C_j^k \tag{16}$$

4. PROPOSED SOLUTION

To fulfill requirements stated in the previous section, in this part, we propose to address the clustering tractography task (described in section 3) as the *hierarchical clustering* based on *dissimilarity approximation*.

4.1 Dissimilarity approximation for tractography

In order to calculate the (dis)similarity bewteen two items, most of the state-of-the-art clustering approaches requires the data to lie in a vectorial space, which is not the case of streamlines. Streamlines are polylines in 3D space and have different lengths and numbers of points. The lack of the vectorial representation avoids the use of these algorithms. In this case, the dissimilarity space representation could be the way to provide such a vectorial representation and for this reason it is crucial to assess the degree of approximation introduced in [22].

The dissimilarity representation is an Euclidean embedding technique defined by selecting a set of objects (e.g. a set of streamlines) called prototypes, and then mapping any new object (e.g. any new streamline) to the vector of distances from the prototypes. Let have an N-streamline set $\mathcal{T} = \{s_1, \ldots, s_N\}$. Let $d : \mathcal{T} \times \mathcal{T} \mapsto \mathbb{R}^+$ be a distance function between two streamlines in $\mathcal{T}$. Note that d is not assumed to be necessarily metric. Let $\Pi = \{\tilde{s}_1, \ldots, \tilde{s}_p\}$, where $\forall i\ \tilde{s}_i \in \mathcal{T}$ and p is finite. We call each $\tilde{s}_i$ as *prototype* or *landmark*. The *dissimilarity representation/projection* is defined as $\phi^d_\Pi(s) : \mathcal{T} \mapsto \mathbb{R}^p$ s.t.

$$\phi^d_\Pi(s) = [d(s, \tilde{s}_1), \ldots, d(s, \tilde{s}_p)] \tag{17}$$

and maps a streamline s from original space $\mathcal{T}$ to a vector of $\mathbb{R}^p$. We define the distance between projected objects as the Euclidean distance between them: $\Delta^d_\Pi(s, s') = ||\phi^d_\Pi(s) - \phi^d_\Pi(s')||_2$, i.e. $\Delta^d_\Pi : \mathcal{T} \times \mathcal{T} \mapsto \mathbb{R}^+$. Obviously, Δ^d_Π and d should be strongly related. Two main issues of the dissimilarity approximation lie on the strategies to choose prototypes (the number of prototypes and how to choose prototypes) and the measure of approximation to estimate the level of reconstruction $\mathcal{T}$ from $\phi^d_\Pi(\mathcal{T})$ (usually called *lossy*).

The issue of choosing the prototypes in order to achieve the desired degree of approximation usually bases on three popular selection policies: random selection, farthest first traversal (FFT) and subset farthest first (SFF) [27]. *Random policy* draws uniformly at random from $\mathcal{T}$, i.e. $\Pi \subseteq \mathcal{T}$ and $|\Pi| = p$. *FFT* selects an initial prototype at random from $\mathcal{T}$ and then each new one is defined as the farthest element of $\mathcal{T}$ from all previously chosen prototypes. FFT policy is related to the *k-center* problem : given a set S and an integer k, what is the smallest ϵ for which you can find an ϵ-cover[3] of S of size k? *SFF* policy samples $m = \lceil cp \log p \rceil$ points from $\mathcal{T}$ uniformly at random and then applies FFT on this sample in order to select the p prototypes (c is an constant). All these policies are parametric with respect to the number of prototypes p. In the case of tractography, based on our experiments, SFF obtains an efficient and effective selection of the prototypes compared with two other methods [22]. As a measure of the degree of approximation of the dissimilarity representation, in the literature of the Euclidean embeddings of metric spaces, the term of *distortion* is usually used [14]. It represents the relation between the distances in the original space and in the projected space. The embedding is said to have *distortion*$\leq c$ if for every $s, s' \in \mathcal{T}$:

$$d(s, s') \geq \Delta^d_\Pi(s, s') \geq \frac{1}{c} d(s, s'). \tag{18}$$

However, this embedding is computationally too expensive to be used in practice. We investigate the relationship between the distribution of distances among streamlines in $\mathcal{T}$ through d and the corresponding distances in the dissimilarity representation space through Δ^d_Π. A good dissimilarity representation must be able to accurately preserve the partial order of the distances, i.e. if $d(s, s') \leq d(s, s'')$ then $\Delta^d_\Pi(s, s') \leq \Delta^d_\Pi(s, s'')$ for each $s, s', s'' \in \mathcal{T}$. We define the Pearson correlation ρ between the two distances over all possible pairs of streamlines in $\mathcal{T}$:

$$\boldsymbol{\rho} = \frac{\mathrm{Cov}(d(s, s'), \Delta^d_\Pi(s, s'))}{\sigma_{d(s,s')} \sigma_{\Delta^d_\Pi(s,s')}} \tag{19}$$

An accurate correlation coefficient between objects in $\mathcal{T}$ results in values of $\boldsymbol{\rho}$ far from zero and close to 1. The correlation focusses on the *averaged* differences between the original and projected space while the distortion cares about the worst case scenario. For this reason, in our case, the correlation is a more appropriate measure.

4.2 Hierarchical clustering

Hierarchical clustering [12] produces a structure of clusters that is more informative than the unstructured set of clusters returned by flat clustering. This characteristic meets the requirement of creating an m-partition set $\mathbb{P}$ of $\mathcal{T}$ without re-running the clustering algorithm again. Another advantage is that hierarchical clustering does not require to pre-specify the number of clusters. It builds nested clusters by merging them successively,

[3] Given a metric space $(\mathcal{X}, d)$, for any $\epsilon > 0$, an ϵ-cover of a set $S \subset \mathcal{X}$ is defined to be any set $T \subset S$ such that $d(x, T) \leq \epsilon, \forall x \in S$. Here $d(x, T)$ is the distance from point x to the closest point in set T

and this hierarchy of clusters represented as a tree (or dendrogram). The root of the tree is the unique cluster that gathers all the samples, the leaves being the clusters with only one sample. In another way, hierarchical clustering algorithm can clusters data firstly on N centers and consequently until only one center. This main character leads to the capability of visualizing $\mathcal{T}$ in many levels of abstraction, and the users can browse the value of level from 1 to N, to see the clusters immediately.

With an N-streamline set $\mathcal{T} = \{s_1, \ldots, s_N\}$, hierarchical produces Q clusters of $\mathcal{T}$ $H = \{H_1, \ldots, H_Q\}$, with $Q \leq N$, such that $C_i \in H_m, C_j \in H_l, m > l$ imply $C_i \in C_j$ or $C_i \cap C_j = \varnothing$, for all $i, j \neq i, m, l = 1, \ldots, Q$. Hierarchical clustering algorithms are either top-down or bottom-up. Bottom-up algorithms treat each streamline as a singleton cluster at the outset and then successively merge (or agglomerate) pairs of clusters until all clusters have been merged into a single cluster that contains all tracts. Bottom-up hierarchical clustering is therefore called Hierarchical Agglomerative Clustering (HAC). Top-down clustering requires a method for splitting a cluster. It proceeds by splitting clusters recursively until individual streamlines are reached [12].

Given an N-streamline set $\mathcal{T}$ and an $N \times N$ (dis)similarity matrix, the basic process of HAC clustering [12] is this:

1. Assign each streamline to one cluster. We now have N clusters, each containing just one streamline.
2. Find the closest (most similar) pair of clusters and merge them into a single cluster.
3. Compute distances (similarities) between the new cluster and each of the old clusters.
4. Repeat steps 2 and 3 until all streamlines are clustered into a single cluster of size N.

Step 3 can be done in different ways, which distinguishes single-linkage, complete-linkage and average-linkage. In *single-linkage* clustering (also called the connectedness or minimum method), the distance between a pair of clusters A and B is the shortest distance from any streamline of one cluster to any streamline of the other cluster.

$$d(A, B) = \min_{s_A \in A, s_B \in B} d(s_A, s_B) \tag{20}$$

where $d(s_A, s_B)$ is the distance between two streamlines:

$$d(s_A, s_B) = \min_{x_i^A \in s_A, x_j^B \in s_B} ||x_i^A - x_j^B||_2 \tag{21}$$

with x_i^A and x_j^B are points belonging to streamline s_A and s_B respectively. In *complete-linkage* clustering (also called the diameter or maximum method), we consider the distance between cluster A and cluster B to be equal to the greatest distance from any member of one cluster to any member of the other cluster.

$$d(A, B) = \max_{s_A \in A, s_B \in B} d(s_A, s_B) \tag{22}$$

In *average-linkage* clustering, the distance between two clusters A and B is defined as the average distance from any streamline of cluster A to any streamline of cluster B.

$$d(A, B) = avg_{s_A \in A, s_B \in B} d(s_A, s_B) \tag{23}$$

In this project, we use the bottom-up hierarchical clustering (HAC strategy) combining with many distance measurements. The reason it that it is readily available to the fact that at the lowest level of abstraction, all streamlines should be presented, and when user changes the level of abstraction, some of present streamlines would be grouped and replaced by a representative. We hope that the HAC clustering can generate meaningful clusters with minimum memory consumption and respond in seconds with changes from user. However, HAC has not the capability to solve the problem of *neighbor checking*. The solution for this problem is still under the investigation.

5. PRELIMINARY RESULTS

As described in figure 2, the proposed framework for tract segmentation has two main steps: creating hypo candidate from whole brain tractogprahy base on supervised learning, and refining candidate using unsupervised learning. How to create tractography from the raw dMRI data can be found in our technical report [1]. The hypo generation step is inherited from the work in [21], and our main investigation in this project focuses on the refinement step. In this part, we only present some preliminary results involved to the refinement step. First, the investigation of the dissimilarity approximation for tractography will be presented in section 5.1. The next section 5.2 describes Spaghetti, a streamline interaction visualization tool, which is the implementation of the framework in figure 2. The last section 5.3 shows a clinical application of the segmentation on finding the difference of CST [4] between healthy and ALS-diseased brains [5].

5.1 Tractography dissimilarity approximation

In this section we describe the assessment of the degree of approximation of the dissimilarity representation across different prototype selection policies and different numbers of prototypes [22]. The aim is to investigate the trade-off between accuracy and computational cost. The robustness of the method is checked first using simulated data, and then on the real tractography. But in the following, we only describe the experiment on real dMRI data. More detail about these experiments can be found in our recent publication [22] at PRNI-2012 [6]

We estimated the dissimilarity representation over real tractography data from dMRI recordings at the Cognition and Brain Sciences Unit, Cambridge, United Kingdom. The dataset consisted of 12 healthy subjects; 101 (+1, i.e. $b = 0$) gradients; b-values from 0 to 4000; voxel size: $2.5 \times 2.5 \times 2.5mm^3$. In order to get the tractography, we used the deterministic tracking algorithm [7]. We obtained two tractographies using 10×10^3 and 3×10^6 random seeds respectively. The first tractography consisted of approximately 10^3 streamlines and the second one of 3×10^5 streamlines. We used all three policies of choosing prototype: random, FFT and SFF. An example of a set of prototypes from 3×10^5 streamlines is shown in figure 3. As the distance between streamlines we chose the most common one, i.e. the mean average minimum (MAM) distance from [32] defined as $d_{mam}(s, s') = \frac{1}{2}(\delta(s, s') + \delta(s, s'))$ where

$$\delta(s, s') = \frac{1}{|s|} \sum_{\mathbf{s}_i \in s} \min_{\mathbf{s'_j} \in s'} ||\mathbf{s}_i - \mathbf{s'}_j||_2. \tag{24}$$

As shown in figure 4(left), in the case of the 10^3-streamline tractography, both FFT and SFF ($c = 3$) had significantly higher

[4] Corticol Spinal Tracts: http://dti-challenge.org/

[5] Amyotrophic Lateral Sclerosis: http://www.alsa.org/

[6] http://www.mlnl.cs.ucl.ac.uk/prni2012/

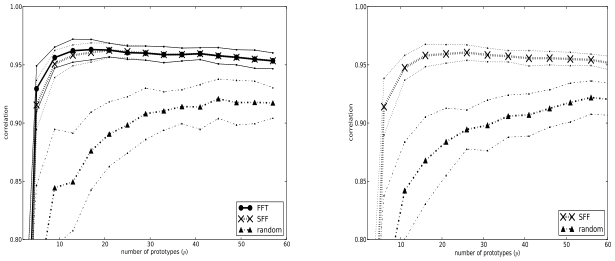

Figure 4: The correlation between of d and Δ_{Π}^{d} over a tractography for different prototype selection policies: $\mathbf{10^3}$ streamlines (left) and $\mathbf{3 \times 10^5}$ streamlines (right).

correlation than the random sampling for all numbers of prototypes considered. We confirmed that the SFF selection policy is an accurate approximation of the FFT policy for tractographies. Moreover we noted that after $15-20$ prototypes the correlation reached approximately 0.95 on average (60 repetitions) and then slightly decreased indicating that a little number of prototypes was sufficient to reach a very accurate dissimilarity representation. Figure 4(right) shows the correlation between SFF and the random policy when the tractography has 3×10^5 streamlines, i.e. the standard size of a tractography from current dMRI recording techniques. In this case, FFT was impractical to be computed because it required approximately 15 minutes on a standard desktop computer for a single repetition when $p = 60$. The cost of computing SFF was instead the same of the case of 10^3 streamlines, as its computational cost depended only on the number of prototypes. It took ≈ 2 seconds on standard desktop computer when $p = 60$ to compute one repetition. We observed that for 3×10^5 streamlines, SFF significantly outperformed the random policy and reached the highest correlation of 0.96 on average (60 repetitions) for $15 - 25$ prototypes. Note that the figures presented in this section refers to data from subject 1 of the dMRI dataset. We conducted the same experiments on other subjects obtaining equivalent results.

All of the results from both simulated data and real tractography data reached correlation ≥ 0.95, showing a strong evident that dissimilarity approximation works well for preserving the relative distances. We advocate that the dissimilarity representation can produce compact feature spaces for the tractography. Moreover we strongly suggest the use of the SFF policy to obtain an efficient and effective selection of the prototypes.

5.2 Spaghetti: an interaction visualization tool for tract segmentation

In this part, we present a streamline interaction visualization tool, called Spaghetti, which is the implementation of the framework in figure 2. But until now, we have not integrated the hypo generation step in this tool yet. It is supposed to use the whole tractography $\mathbb{T}$ instead of a set of candidates $\mathcal{T}$. The process that we propose works recursively: starting from a small number of clusters of streamlines the user decides which clusters to explore. Exploring a cluster means that the application re-clusters its content at a finer grained level or provides the content (means streamlines) of that cluster. In another words, the users change the level of abstraction and we provide cluster representatives of new clusters according to that level.

As the distance between streamlines, beside the MAM (equation 24, we also use the MDF (minimum average direct flip)

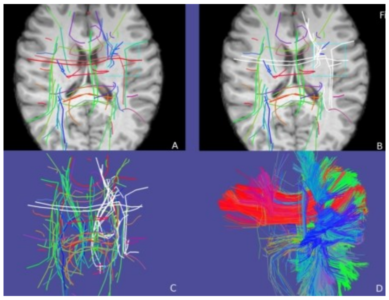

Figure 5: Streamline interaction with Spaghetti: the initial representatives (A); some selected clusters in white color (B); only representatives without slices (C); exploring representatives by hundred of streamlines (D)

$$d_{mdf}(s,s') = min(\delta_{direct}(s,s'), \delta_{flipped}(s,s')) \qquad (25)$$

with $\delta_{direct}(s,s') = \frac{1}{k}\sum_{i=1}^{k} ||\mathbf{x}_i - \mathbf{x'}_i||_2$. and $\delta_{flipped}(s,s') = \frac{1}{k}\sum_{i=1}^{k} ||\mathbf{x}_i - \mathbf{x'}_{k-i}||_2$. where k is the number of points $\mathbf{x}_i$ on the two tracks s and s'. In this software, currently we use the fast clustering algorithm proposed in [8], called QuickBundle(QB). The reason is that QB is very simple, easy to implement and very fast. However, every time user change the level of abstraction, we need to re-run QB to get the new clusters. It is one of the drawbacks of the current version of Spaghetti.

The application starts visualizing the brain as a set of a few cluster representative. Each representative track acts as the access point to the streamlines within that cluster and allows the user to address this portion of the tractography as a single unit, which we call "bundle of interest" (BOI). After visually inspecting the simplified tractography the user interactively selects one or more representative tracks and explores them. In order to explore the detailed of the selection, the user may ask to re-cluster the selected BOIs, and possibly further refine the initial selection. After selecting one or more of the small clusters through their representatives the user can repeat the visual inspection step, and the re-clustering step as required in order to unveil the local structures. Beside the capability of interacting with streamlines, in this tool we also provide function of visualizing slices of structural volumes (see figure 5). The volume is aligned in the same space (e.g standard space) of the tractography. This enables medical practitioners and researchers to meaningfully navigate the entire space of the tractography related to anatomy. The detail of this tool can be found in our publication [9] at OHBM-2012[7]

5.3 Clinical diagnosis application: the difference of Corticol Spinal Tracts between a healthy and an ALS-diseased brain

In this part, we try to demonstrate the usefulness of tract segmentation in clinical study. The aims of this work are to: first, finding the differences of the Corticol Spinal Tracts (CST) between a healthy brain and an ALS (Amyotrophic Lateral Sclerosis) diseased brain based on tract quantification; and second present a general framework for clinical diagnosing based on the differences between two folders of interesting tracks.

ALS, also known as motor neurone disease or Lou Gehrigs

[7] Organization for Human Brain Mapping 2012 `http://www.humanbrainmapping.org/OHBM2012/`

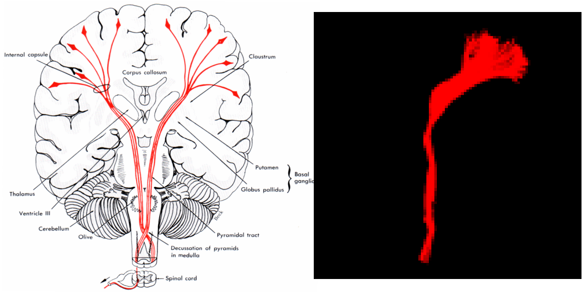

Figure 6: Left - the Cortico Spinal Tracts in general. Right - the left CST segmentation of the control 201 in the dataset ALS

disease, is a progressive neurodegenerative disease that affects nerve cells in the brain and in the spinal cord. As motor neurons degenerate, they can no longer send impulses to the muscle fibers that normally result in muscle movement. The result is wasting and atrophy of muscles, leading to difficulties in speaking, swallowing, stumbling, etc. Because ALS disease involves to the nerve cells in ***CST***, in this part, we only focus on this CST tract. Usually, CST starts from the cerebral cortex, and terminates in the spinal cord. Note that fibers after crossing over from one side to the other continue downward in the lateral corticospinal tract on the opposite side and go to muscles (see figure 6-left). That is the reason why impulses from one side of the cerebrum cause movements of the opposite side of the body.

CST segmentation was done by doctors using Spaghetti tool on the ALS dataset, recorded with a $3T$ scanner at Utah Brain Institute. This dataset consisted of 12 healthy controls and 12 subjects; 64 (+1, i.e. $b = 0$) gradients; b-values 1000; voxel size: $1. \times 1. \times 1.mm^3$. For creating tractography, we used the same algorithm in section 5.1, with 3×10^5 random seeds. An example of CST segmentation from ALS dataset is in figure 6 (left). As the result, we had 48 segmentations (24 of patients including 12 left CST and 12 right CST; and similar for controls).

From the prior knowledge of neuroscientists and doctors, there is an evidence about the reducing of the number of fibers in CST of ALS patients compared with control people. It is also the same situation with the volume of CST. Beside, fractional anisotropy (FA) and mean diffusion (MD) also play an important role for recognizing the ALS disease. Following are some quantitative features which may effectively affect on ALS patients: *fiber count* - the number of streamlines belonging to CST; *fiber length* in *mm* (min, max, mean length); *fiber volume* - number of voxels occupied by all streamlines or the bounding geometry cylinder of CST; *fiber density* - ratio between fiber count and voxel number; *fragmentation* - ratio between fiber count and the volume; *fractional anisotropy (FA)* - defined as mean value of the standard deviation in the three eigenvalues and in the range 0 to 1

$$FA = \frac{1}{\sqrt{2}} \frac{\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_3 - \lambda_1)^2}}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \tag{26}$$

and *mean diffusion (MD)* - the average diffusion rate in all directions

$$MD = \frac{trace(DT)}{3} = \frac{\lambda_1 + \lambda_2 + \lambda_3}{3} \tag{27}$$

where $(\lambda_1, \lambda_2, \lambda_3)$ is eigenvalues of diffusion at a given voxel.

After calculating the value of these features, we did a $t - test$ on each set of left and right. It showed that, in the right CST, fiber number significantly decreased ($p = 0.00042$) in ALS patients ($mean_{fiber-number} = 294$) compared with controls ($mean_{fiber-number} = 640$). In contrast, patients had the fiber min length slightly higher then controls ($p = 0.07$, patients: $mean_{min-length} = 74.25$ and controls: $mean_{min-length} = 53.9$). Moreover, the volumn of the left CST dramatically diminished ($p = 0.0034$) between patients ($mean_{volumn} = 6038$) and healthy peoples ($mean_{volumn} = 4230$). These are just some preliminary results and it needs more investigation to confirm the difference between healthy and ALS-diseased brain. But it also shows an strong evidence that the tract segmenation has a bright capability for applying in clinical diagnose application.

6. CONCLUSION AND FUTURE WORKS

In this document we investigate the using of machine learning for tractogarphy segmentation task. We design the framework of interactive segmentation based on two steps: tract identification and tract refinement. The first step produces the the initial candidate from a full tractography by supervised learning. The second one aims to refine the candidate by removing selected streamlines or adding more ones, and is conceived as a clustering task. Our main focus is on this step rather than the previous one, which can be done by using the method proposed in [21].

First, we propose a solution for clustering tractography based on Hierarchical clustering, which meets many requirements in our case. Second, we study the dissimilarity approximation for tractography, to present streamlines in a vectorial space. We also provide an implementation of our proposed two-step framework for tract segmentation, a streamline interaction tool, called Spaghetti. In this scientific interaction tool, we present an simple way to interact and segment streamlines in $3D$ space, which goes, as far as we know, beyond any other available medical imaging software. This enables medical practitioners and researchers to meaningfully navigate the entire space of the tractography and perform the segmentation task more easily and accuracy.

However, up to present, we still represent streamlines in the original space (a polyline $s = \{\vec{x_1}, \ldots, \vec{x}_{n_s}\}$, where $\vec{x} \in \mathbb{R}^3$). This leads to a fact that the streamline distance(*mam* 24 or *mdf* 25) sometimes is not a *real distance*. For example, the distance between a very short streamline (n_s small) and a long streamline (n_s large) is really large, no matter what two streamlines are far each other or not. This drawback should be dismissed when we represent streamline in the dissimilarity approximation. Converting from original space into dissimilarity approximation space, and define a new distance measurement is one of our future works. Beside, in the current version of Spaghetti, we use QB to cluster streamlines. Although we propose HAC clustering and point out many advantages of it in our case, but we have not integrated it into Spaghetti. In a near future, QB should be replaced by HAC. Moreover, Hierarchical creates a tree of nested cluster, it also needs to design a suitable structure for storing and accessing these clusters as well. Beside, we have not found the solution for the *neighbor checking* problem yet. It is also another future work. About the clinical diagnosis application, we believe that the satisfactory result will be published in near future. After that, the general framework for clinical diagnosing based on the differences between two folders of interesting tracks must be presented and applied for other brain diseases.

7. REFERENCES

[1] N. T. Bao, E. Garyfallidis, and E. Olivetti. dMRI pre-processing: from raw data to coregistered tractographies. Technical report, Feb. 2012.

[2] P. J. Basser, J. Mattiello, and D. LeBihan. MR diffusion tensor spectroscopy and imaging. *Biophysical journal*, 66(1):259–267, Jan. 1994.

[3] M. Bozzali, A. Falini, M. Franceschi, M. Cercignani, M. Zuffi, G. Scotti, G. Comi, and M. Filippi. White matter damage in Alzheimer's disease assessed in vivo using diffusion tensor magnetic resonance imaging. *Journal of Neurology, Neurosurgery & Psychiatry*, 72(6):742–746, June 2002.

[4] A. Brun, H. Knutsson, H.-J. Park, M. E. Shenton, and C.-F. Westin. Clustering Fiber Traces Using Normalized Cuts. pages 368–375. 2004.

[5] R. Caruana and A. Mizil. An empirical comparison of supervised learning algorithms. In *Proc. of the 23rd Intl. conf. on Machine learning*, ICML '06, pages 161–168, NY, 2006. ACM.

[6] M. P. Dubuisson and A. K. Jain. A modified Hausdorff distance for object matching. In *Proceedings of 12th International Conference on Pattern Recognition*, volume 1, pages 566–568. IEEE Comput. Soc. Press, 1994.

[7] E. Garyfallidis. *Towards an accurate brain tractography.* PhD thesis, University of Cambridge, 2012.

[8] E. Garyfallidis, M. Brett, M. M. Correia, G. B. Williams, and I. Nimmo-Smith. QuickBundles, a method for tractography simplification. *Frontiers in Neuroscience*, 6(175), 2012.

[9] E. Garyfallidis, S. Gerhard, P. Avesani, T. B. Nguyen, V. Tsiaras, I. Nimmo-Smith, and E. Olivetti3. A software application for real-time, clustering-based exploration of tractographies. 2012.

[10] G. Gerig, S. Gouttard, and I. Corouge. Analysis of brain white matter via fiber tract modeling. In *In: Proc. IEEE Int. Conf. EMBS. 2004*, volume 2, pages 4421–4424, 2004.

[11] Z. Ghahramani. Unsupervised Learning. In *Advanced Lectures on Machine Learning*, pages 72–112. 2004.

[12] S. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, Sept. 1967.

[13] L. Jonasson, P. Hagmann, J. Thiran, and V. Wedeen. Fiber tracts of high angular resolution dMRI are easily segmented with spectral clustering. In *ISMRM*, 2005.

[14] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, June 1995.

[15] M. Maddah, A. U. J. Mewes, S. Haker, Grimson, and S. K. Warfield. Automated Atlas-Based Clustering of White Matter Fiber Tracts from DTMRI. In J. S. Duncan and G. Gerig, editors, *MICCAI 2005*, volume 3749 of *Lecture Notes in Computer Science*, chapter 24, pages 188–195. Springer, Berlin, Heidelberg, 2005.

[16] M. Maddah, L. Zöllei, E. E. Grimson, and W. M. Wells. Modeling of Anatomical Information in Clustering of White Matter Fiber Trajectories Using Dirichlet Distribution. *IEEE on Pattern Analysis and Machine Intelligence.*, 2008:1–7, July 2008.

[17] B. Moberts, A. Vilanova, and J. J. van Wijk. Evaluation of Fiber Clustering Methods for Diffusion Tensor Imaging. In *IEEE Visualization*, pages 65–72, 2005.

[18] S. Mori and P. C. M. van Zijl. Fiber tracking: principles and strategies, a technical review. *NMR Biomed.*, 15(7-8):468–480, 2002.

[19] R. Neji, A. Besbes, N. Komodakis, J. Deux, M. Maatouk, A. Rahmouni, G. Bassez, G. Fleury, and N. Paragios. Clustering of the human skeletal muscle fibers using linear programming and angular Hilbertian metrics. *Info. processing in Med. Imaging*, 21:14–25, 2009.

[20] L. J. O'Donnell and C.-F. F. Westin. Automatic tractography segmentation using a highdimensional white matter atlas. In *IEEE Trans. Med. Imag*, pages 1562–1575, 2007.

[21] E. Olivetti and P. Avesani. Supervised segmentation of fiber tracts. In *Proceedings of SIMBAD'11*, SIMBAD'11, pages 261–274, Berlin, Heidelberg, 2011. Springer-Verlag.

[22] E. Olivetti, T. B. Nguyen, and E. Garyfallidis. The Approximation of the Dissimilarity Projection. *Pattern Recognition in NeuroImaging, IEEE International Workshop on*, 0:85–88, 2012.

[23] E. Olivetti, S. Veeramachaneni, S. Greiner, and P. Avesani. Brain connectivity analysis by reduction to pair classification. In *Proceeding of Cognitive Information Processing (CIP), 2010*, pages 275–280, June 2010.

[24] P. Rai and S. Singh. A Survey of Clustering Techniques. *International Journal of Computer Applications*, 7(12), Oct. 2010.

[25] P. Savadjiev, J. Campbell, G. Pike, and K. Siddiqi. Streamline Flows for White Matter Fibre Pathway Segmentation in Diffusion MRI. pages 135–143. 2008.

[26] D. S. Tuch, T. Reese, M. Wiegell, N. Makris, J. Belliveau, and V. Wedeen. High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. *Magn. Reson. Med.*, 48(4):577–582, Oct. 2002.

[27] D. Turnbull and C. Elkan. Fast recognition of musical genres using RBF networks. *IEEE Trans. on Knowledge and Data Engineering*, 17(4):580–584, Apr. 2005.

[28] S. Wakana, A. Caprihan, M. Panzenboeck, J. Fallon, M. Perry, R. Gollub, K. Hua, J. Zhang, H. Jiang, P. Dubey, A. Blitz, P. Zijl, and S. Mori. Reproducibility of quantitative tractography methods applied to cerebral white matter. *NeuroImage*, 36(3):630–644, July 2007.

[29] X. Wang, Grimson, and C.-F. Westin. Tractography segmentation using a hierarchical Dirichlet processes mixture model. *NeuroImage*, 54(1):290–302, Jan. 2011.

[30] D. Wassermann, L. Bloy, E. Kanterakis, R. Verma, and R. Deriche. Unsupervised white matter fiber clustering and tract probability map generation: applications of a Gaussian process framework for white matter fibers. *NeuroImage*, 51(1):228–241, May 2010.

[31] R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Trans. on neural networks*, 16(3):645–678, May 2005.

[32] S. Zhang, S. Correia, and D. H. Laidlaw. Identifying White-Matter Fiber Bundles in DTI Data Using an Automated Proximity-Based Fiber-Clustering Method. *IEEE Transactions on Visualization and Computer Graphics*, 14(5):1044–1053, Sept. 2008.

[33] O. Zvitia, A. Mayer, and H. Greenspan. Adaptive mean-shift registration of white matter tractographies. In *the 5th IEEE Intl. Symposium on Biomedical Imaging, 2008. ISBI 2008.*, pages 692–695, May 2008.