UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Corso di Laurea Magistrale in Bioingegneria

TESI DI LAUREA

Anatomical comparison between Gilles de la Tourette syndrome patients and control subjects based on the framework of currents

Laureando: **Pietro Gori**

Relatore Padova: **Prof. Maria Elena Valcher**

Co-relatore DTU: **Prof. Line Katrine Harder Clemmensen**

Co-relatore ICM: **Prof. Stanley Durrleman**

Anno Accademico 2011-2012

Summary

Gilles de la Tourette syndrome (GTS) is a childhood-onset chronic neuropsychiatric disorder characterised by multiple motor tics and vocal tics which lead to significant social and educational impairments. Several studies ([Kim10],[MM11],[TM09],[Sin05]) show an increase of the prevalence of this syndrome in the last decades. This discovery has brought many research teams to focus their studies on this syndrome. The diagnosis is still based predominantly on the history and observation of the tics and the causes and the development of this disease are still unclear.

The data set of this project is composed by cortical surfaces, sub-cortical structures and neural fiber bundles segmented from MRI and DWI images belonging to two populations: one of controls and another one of patients subject to GTS. The goal of this Master's Thesis, associated to a future PhD project, is to understand if there are anatomical differences between these two populations especially in the main neuronal circuits described in Worbe et al. [WGH+10] (cortico-striato-thalamo-circuits).

The recent development of a new framework based on mathematical objects called "currents" seems very promising in this regard [Dur10]. Currents can represent both surfaces (cortical surfaces, sub-cortical structures) and curves (fiber bundles) and it's also possible to compare them without the need of any kind of correspondence.

In the spirit of the method proposed in Durrleman et al. [DPK+12], an atlas is built, namely a template shape, for each kind of object (surfaces and curves) present in the population. Each template captures the anatomical invariant of the population and it is then deformed to each subject's object using a parametrizable smooth, one-to-one, non-linear deformation of the underlying 3D space (diffeomorphism). Every template is deformed in the same 3D space but each deformation is characterized by different parameters. These deformations represent the variability of the anatomical objects within the population.

This methodology was tested only on surfaces using a gradient descend scheme which allows to estimate both the templates and the deformations while preserving the topology of the initial template given by the user. One of the main contributions of this project is to adapt this methodology also for fiber bundles. From a geometrical point of view fiber bundles can be seen as sets of curves. The connectivity between points is exactly what we want to keep. This permits to retrain information about the starting and ending point of the bundles and therefore also about the linked areas in the cortical surface and internal structures. This allows us to detect the parts of the main neuronal circuits more variable between the two populations.

We propose also a generic method to initialize the bundle template by selecting the most representative fibers among all the subjects present in the population. This atlas construction method applied on both populations will give as result a template for each kind of object with its relative parametrizable deformations to each object. Analysing both the templates and their main variability (i.e. PCA), it will be possible to understand if there are anatomical differences between the two populations and above all where they are present. We will use this information to test the hypothesis described before of Worbe et al. [WGH+10] about Gilles de la Tourette syndrome.

Acknowledgements

First of all I would like to thank my 3 supervisors:

- Merci beaucoup Stanley pour m'avoir introduit dans la communité de Shape Analysis, pour avoir partagé avec moi ta connaissance et surtout pour m'avoir montré comment bien exercer le métier de chercheur.

- Thank you very much Line for finding this beautiful laboratory in Paris and for your constant help.

- Grazie mille Professoressa Valcher soprattutto per il suo aiuto durante i due anni in Danimarca. E' difficile trovare una professoressa altrettanto gentile ed efficiente quanto lei.

Moreover I am truly grateful to Linda Marrakchi-Kacem and Cyril Poupon from the NeuroSpin Institute for sharing with me their database. This project wouldn't have been possible without their aid.
I am also thankful to Yulia Worbe and Andrea Hartmann (from the team of Marie Vidailhet and Stéphane Lehéricy at CR-ICM) for letting me work with their data base about GTS patients. This dataset is extremely interesting and I understood immediately how lucky I was. Therefore I want to thank you again and I hope that our collaboration will become closer during the PhD.
After I would like to express my gratitude also to Olivier Colliot for accepting me in the Cogimage group. His leading ability and scientific knowledge was outstanding, showing always an interest in all the different projects.
Furthermore I want to thank also all the members of the Cogimage group. It is a wonderful and friendly group composed by people who supported me and helped me during these five months. Grazie a tutti!

Contents

Introduction

In the last decades, with the advances in digital imaging and computation, more and more researchers focused their studies on the biological variability of human anatomy. This kind of study can be traced back to the beginnings of modern science. In fact already Albrecht Dürer studied the variability of faces in his work: *"Vier Bücher von Menschlicher Proportion"* in 1528, where he also used the works of Leonardo da Vinci. He chose some landmarks "a priori" like the end of the nose or the beginning of the forehead to study how they can vary among a population of about 200-300 living persons. Moreover he also applied the mathematical simulation of convex and concave mirrors to deform some faces studying the results. But the aim of this deformation was mainly physiognomy (i.e. the study of the character of a person from his outer appearance). Only in 1917 D'Arcy Thompson with his book *"On Growth and Form"* had a clear vision of what would happen more than 70 years later. He understood that the essential task of morphology is not to define in a precise way each shape but to find a comprehensive way to compare them. He emphasized the fact that the study of the variability among a population of shapes could be easier to understand than the study of each figure itself. Thus, looking at the deformations of a "common average shape" (i.e. template) into every different shape can bring more information than the shapes themselves.

Thompson also underlined the importance of Mathematics, above all the "Method of Coordinates" on which is based the Theory of Transformation [Mil04]. Thompson's vision is exactly the mathematical structure being studied in *Computational Anatomy* [GM98] that presents three main aspects [MQ09]:

1. Automated construction of anatomical manifolds $\in \mathbf{R}^3$ such as points, curves, surfaces, or volumes.

2. Comparison of these manifolds.

3. Statistical codification of the variability of anatomy between subjects for testing hypothesis, for example, related to disease states.

This project is mainly based on the two last points. The extraction of 3D curves and 3D surfaces from T1-weighted MRI and DWI images was carried out by Linda Marrakchi-Kacem in her PhD Thesis [MK11].

The problem of comparing two different structures such as curves or surfaces received tremendous focus by many research teams and it is usually achieved by matching the two objects. This means transforming (i.e. deforming) the first object such that it is as similar as possible to the second one and eventually studying the transformation.

In order to analyse the similarity between two objects, one needs a definition of distance. This dissimilarity measure should not be limited to a particular kind of data since, like in this project, one wants to process both curves and surfaces or a mix of them. Furthermore it should not rely on any "a priori" knowledge such as a correspondence between distinct parts of the two objects (i.e. landmarks) chosen by the researchers. In fact, this can bias the result making it also highly dependent on the choice of the landmarks. Moreover it should also be weakly sensitive to the sampling of the objects (i.e. when they are characterized by different amount of points). A framework which fulfils all these requirements is the one of currents. Currents are generalized distributions which represent the geometrical objects such as curves or surfaces in 3D. They are equipped with an Hilbert Space structure and an easily computable norm. The norm of the difference between two currents (i.e. representations of curves or surfaces in 3D) is exactly the measure of distance used throughout this project.

Once it is defined how to represent the geometrical objects and how to compare them, it is necessary to introduce a deformation framework which enables to explain how to deform and match an object to another one. The deformation should localize most of the anatomical differences, which include also non-linear variations, it should be compatible with the modeling of shapes as currents and it should also be possible to compute statistics on it. Among other possible choices it seems that the Large Diffeomorphic[1] Deformation Metric Mapping (LDDMM) framework is particularly adapted for these requirements [Dur10] [Gla05]. In fact, it has already been shown that this deformation framework can be used for the registration of shapes modeled as currents and also that it's possible to infer statistics on it [Dur10].

The aim of this project is to combine these two frameworks to analyze the differences between two populations of curves and surfaces in 3D representing part of the brains of healthy controls and of patients subject to Gilles de la Tourette syndrome. In the perspective of Grenander's approach, we decided to build a template for each population

[1] smooth, one-to-one deformation of the underlying 3D space with smooth inverse.

which can be seen as an average of the objects. This deformable template creates an orbit under the group of diffeomorphisms: a whole family of new objects with similar structure as the template. Therefore one can build a template which shows the common features of the population and deform it towards each object of the population. In this way it's possible to compare the two templates of the populations, which represent the two averages of the objects of the populations, looking for anatomical differences that can be useful for medical studies. Moreover, since the deformation used in this project is parametrizable, it's also possible to analyse the variability of the deformations of each population (i.e. PCA) as suggested by D'Arcy Thompson.

Several medical studies present in the Literature show that there is a connection between the shape of the constituents of the brain and the disease of Gilles de la Tourette (Chapter 2). This has also inspired different hypothesis about the evolution of the Gilles de la Tourette syndrome (GTS) and about the area mostly influenced by it. Among them, we decided to test the one of Worbe et al. explained in [WGH+10] and in Chapter 2. This is mainly based on the link between the shape and size of the cortico-striato-thalamo-cortico circuits (Chapter 2) and the seriousness of the disease.

Thus the main goal of this project is to check this last hypothesis on two populations of healthy and diseased subjects (subject to GTS) using the framework of currents for representing the anatomical objects and a diffeomoprhism of the 3D space to deform the template towards each object. Both the template and the deformations will be analyzed with a visual inspection and statistical tools such as PCA looking for anatomical differences between the two populations. This can be really helpful and useful from a medical point of view since it helps the doctors in understanding which are the parts mostly involved with this syndrome.

Chapter 1

The framework of Currents

In this chapter the framework of currents is presented starting with a practical and more physical definition linked with the Faraday's law of induction. For a more rigorous mathematical definition please see Appendix A.2. It is afterwards explained how it can be used in practice in the context of neuroscience dealing therefore especially with surfaces and curves in 3D. We show that currents can represent both curves and surfaces in 3D and they create a vector space provided with a norm and therefore a metric to compare two different objects. This dissimilarity measure doesn't assume any arbitrary correspondence and it takes into account the local orientation of curves or surfaces. Numerical tools for working with currents in a more efficient way are also presented. First of all how to build a regular lattice where it's possible to approximate the framework of currents going therefore from continuous to discrete coordinates. And after that it's also described a method for obtaining a sparse representation of currents: a Matching Pursuit Algorithm presented for the first time in [Dur10]. Lastly it is shown that this representation it's really good for compression and for a better interpretation of the data, but it has also a drawback: it doesn't keep the topology of the initial object loosing the connectivity between points. This makes impossible the employment of this representation for the initialization of the template in the atlas construction as it will be described at the end of this chapter.

1.1 What is a current?

Let **L** be a piecewise differentiable curve in 3D, it can be seen also as a current via the path-integral equation:

$$L(w) = \int_L w(x)^T \tau(x) d\lambda(x) \tag{1.1}$$

for every square integrable 3D vector field w (i.e. $\int_{-\infty}^{+\infty} |w(x)|^2 dx < \infty$), where $\tau(x)$ is the unit tangent vector of the curve **L** at point x and $d\lambda(x)$ denotes the Lebesgue measure on the curves so that $\int_L d\lambda(x)$ equals the total length of the curve.

Similarly, let **S** be a piecewise differentiable surface, it can be seen also as a current thanks to the flux equation:

$$S(w) = \int_S w(x)^T (u(x) \times v(x)) d\lambda(x) = \int_S w(x)^T \eta(x) d\lambda(x) \tag{1.2}$$

for every square integrable 3D vector field w, where $\eta(x) = (u(x) \times v(x))$ is the unit normal vector of the surfaces S at point x and $(u(x) \times v(x))$ is the orthonormal basis of the tangent plane of the surface S at point x.

The idea of current is to characterize every curve and surface, from now on called "shape", by the collection of the real numbers S(w) and L(w) or, in other words, to probe shapes using vector fields w. A practical interpretation of the concept of *current* is linked with Electromagnetism. In fact, Eq. 1.2 is exactly the *flux equation* present in the *Faraday's law of induction* where $w(x)$ is the magnetic field.

This law states that a time varying magnetic field will induce an electric current in a wire loop. More precisely the induced electromotive force (emf)[1] in a wire loop is proportional to the variations of the flux of a magnetic field passing through the surface delimited by the wire loop. Thus it's possible to induce an *emf* by varying the magnitude of the magnetic field with time or by varying the angle between the magnetic field and the Surface. In fact we assume that the size and shape of the surface enclosed by the wire loop doesn't change with time. This means that we can measure the current within the wire using the flux equation (Eq. 1.2) with $w(x)$ equal to the magnetic field **B** and $\eta(x)$ equal to the normal on the surface enclosed by the wire loop. Thus, measuring the current within the wire for every possible variation of the magnetic field (in this case in space not

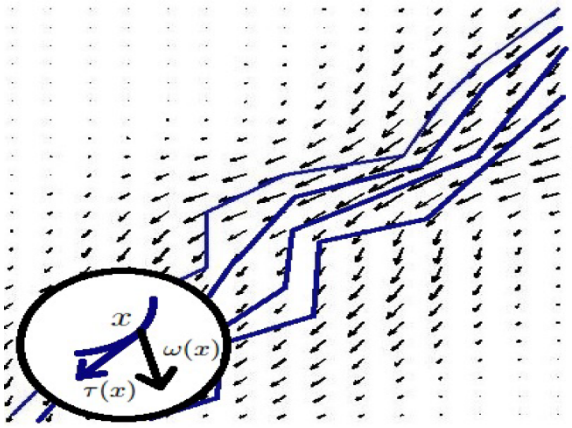
[1] Potential difference between two points, measured in Volts, and it is the cause of charge separation

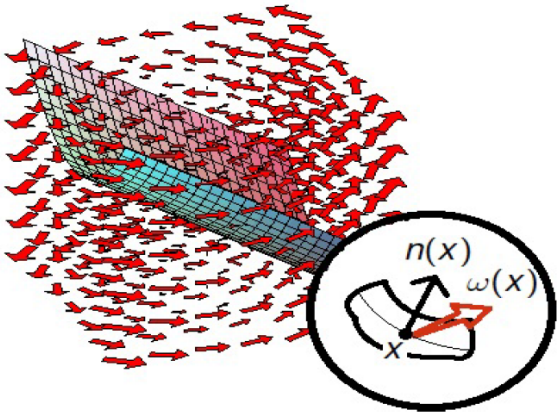
in time), we can then retrieve the geometry of the wire. Or, in other words, we can use the magnetic field to probe the geometry of the wire loop.

Thus, when dealing with surfaces and curves in 3D, the space of currents (denoted as W*) can be defined as the set of continuous linear mappings from a test space of square integrable vector fields W to **R**. And, by definition, W* is exactly the dual space of W. More information about this test space will be given in paragraph 1.3. Every shape can therefore be represented by a current and, being a space of mappings, currents build a *vector space* with its usual properties, denoting T as a generic current:

- $(T_1+T_2)(\mathrm{w})=T_1(\mathrm{w})+T_2(\mathrm{w})$
- $(\lambda T)(\mathrm{w})=\lambda T(\mathrm{w})$
- $(-T)(\mathrm{w})=-T(\mathrm{w}) \rightarrow (T_1-T_1)(\mathrm{w})=T_1(\mathrm{w})-T_2(\mathrm{w})=0$

From these properties one can understand that the response of a sum of shapes (curves or surfaces) is equal to the sum of the responses of each shape. Furthermore that the scaling of a shape by a factor λ is equal to the scaling of the flux equation (or path-integral) by the same factor. And eventually it is important to notice that, considering for example a curve, changing the orientation of the curve means changing the sign of the path integral along the curve and therefore obtaining an opposite current. The union of a curve with itself but with opposite orientation cancels out in the space of current.





$$L(\omega) = \sum_{i=1\ldots4} \int_{L_i} \omega(x)^t \tau(x) d\lambda(x) \qquad S(\omega) = \int_S \omega(x)^t n(x) d\lambda(x)$$

Figure 1.1: At each point x, it is associated a tangent vector $\tau(x)$ (i.e. curves) or normal vector $\eta(x)$ (i.e. surfaces) and the value of the vector field $w(x)$. Using Eq. 1.1 or Eq. 1.2 for all the possible vector fields, one obtains a characterization of the geometrical object. Picture taken from [Dur10].

1.2 Discrete setting

From a practical point of view, when one deals with curves or surfaces, it's useful to shift to a discrete setting. In fact, bundles or fibers from a brain are usually the result of a tractography method, which produces polygonal lines (sets of connected points which build segments). Similarly surfaces in 3D may be seen as triangle mesh which are collections of vertices and faces (triangles) that define the shape of an object (in this project the cortical surface of the brain or the basal ganglia).

Both segments and triangles can be seen as *Dirac delta currents*. In fact all the points along one segment share the same tangent vector, while all the points inside a triangle have the same normal vector. Thus one can approximate the integral along the segment C (Eq. 1.1) by a single value $w(c)^T\tau$ where τ is the tangent vector (difference between last and initial point of the segment) and c is the center of the segment. The same operation could be done also for the triangles where the flux equation (Eq. 1.2) on the triangle S is approximated by a single value $w(c)^T\eta$ where η is the normal of the triangle and c the center. Thus for both cases, segments and triangles in 3D, one can approximate the entire set of N polygonal lines or N triangle mesh by a finite sum of *Dirac delta currents* δ_x^α:

$$T(w) \sim \sum_{k=1}^{N} \delta_{x_k}^{\alpha_k}(w) = \sum_{k=1}^{N} w(x_k)^T \alpha_k \tag{1.3}$$

where x_k are the centres of the segments (or of the triangles) and α_k are respectively the tangents of the segment if dealing with curves or the normals of triangle if one is working with surfaces. The oriented points (x_k, α_k) are called *momenta* and the magnitude of α_k is equal to the length of the segment for curves and the area of the triangle for surfaces.

$$L(w) = \int_L w(x)^T \tau(x) d\lambda(x) \sim \sum_{k=1}^{N} w(x_k)^T \tau_k = \left(\sum_{k=1}^{N} \delta_{x_k}^{\tau_k} \right)(w) \tag{1.4}$$

$$S(w) = \int_S w(x)^T \eta(x) d\lambda(x) \sim \sum_{k=1}^{N} w(x_k)^T \eta_k = \left(\sum_{k=1}^{N} \delta_{x_k}^{\eta_k} \right)(w) \tag{1.5}$$

So $L = \sum_{k=1}^{N} \delta_{x_k}^{\tau_k}$ and $S = \sum_{k=1}^{N} \delta_{x_k}^{\eta_k}$.

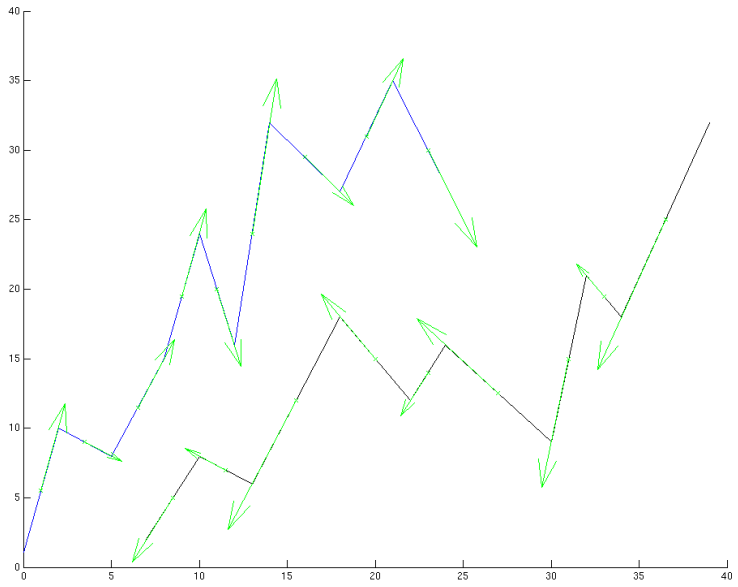


Figure 1.2: Vision of a curve from a discrete point of view as sum of oriented points (momenta) in green, each one positioned at the center of a segment. The axis are unitless on purpose since it's a general example that works with any unit of measure (for space).

1.3 Definition of distance in the metric of currents and RKHS

The space of currents is provided with the following operator norm:

Definition 1.1 Let T be a current. The *mass-norm* of T is defined as the operator norm:

$$M(T) = \sup_{\|w\|_\infty \leq 1} |T(w)| \tag{1.6}$$

Thus, if T is a current in the discrete setting (a sum of Dirac delta current), like $T = \sum_{i=1}^{N} \delta_{x_i}^{\eta_i}$, for N distinct points x_i and vectors η_i; then the *mass-norm* of T is given as: $M(T) = \sum_{i=1}^{N} |\eta_i|$. For the Proof of this Proposition please see [Dur10].

Moreover it can also be demonstrated that the *mass-norm* of the difference of two finite sums of Dirac delta currents located at different points is $M(T - T') = M(T) + M(T')$,

and so, calling $T = \sum_{i=1}^{N} \delta_{x_i}^{\alpha_i}$ and $T' = \sum_{i=1}^{N'} \delta_{y_i}^{\beta_i}$, one could write:

$$M(T - T') = M(T) + M(T') = \sum_{i=1}^{N} |\alpha_i| + \sum_{i=1}^{N} |\beta_i| \tag{1.7}$$

From this equation it turns out that the dissimilarity between two currents (surfaces or curves) is constant as long as the points x_i and y_i are different.

In fact, taking as example two currents composed only by one point each, we will have that $M(\delta_{x_i}^{\alpha_i} - \delta_{y_i}^{\alpha_i}) = 2|\alpha|$ as long as x $\neq$ y. But once x=y the mass-norms is equal to zero and so it shows that this metric is blind to shape dissimilarities until the two shapes are perfectly aligned [Dur10]. In other words it means that the metric captures the shape similarity only if the curves are perfectly superimposed, but if a curve is copied and translated from the original one, the difference will be always constant, independently of the magnitude of the translation. Thus the mass-norm does not seem to be very useful and it is also not well-posed from a numerical point of view due to the discontinuous behaviour of the metric with respect to the positions x and y of the two curves.

A reason of this behaviour comes from the fact that the test space of vector fields doesn't have any spatial smoothness constrain. This means that it's always possible to find a vector field which perfectly interpolates the vectors of the momenta (tangents for curves or normals for surfaces) as long as these curves do not overlap. A vector space that enables to map every square integrable vector field to a smooth one is the one called *Reproducible Kernel Hilbert Space* (RKHS) and it is the set of the convolutions between any square integrable vector field and a smoothing kernel K^W. The Kernel plays the role of a low-pass filter on the spatial frequencies of the vector fields. In this project, the Kernel will be a Gaussian one: $K^W(x, y) = \exp(-|x - y|^2/\lambda_W^2)$Id for any points (x,y), where Id stands for the identity matrix (3x3 in case of a 3D space) while λ_W is the standard deviation of the Kernel and it is one of the most important parameter in this framework. In fact, it is the typical scale at which the vector fields w may vary spatially in this new test space W. This means that it allows to control the metric on the space of currents which means the measure of the distance between curves (or surfaces).

By construction this new space W has two important properties:

1. W is the dense span of the vector fields of the form $w(x) = K^W(x, y)\beta$ for any fixed point y and vector β. This means that any vector field w can be written as an infinite linear combination of the *basis elements* $K^W(x, y)\beta$.

2. W is provided with an inner product which is defined on these basis vectors by

$$\left\langle K^W(\cdot,x)\alpha, K^W(\cdot,y)\beta \right\rangle = \alpha^T K^W(x,y)\beta \tag{1.8}$$

Denoting with w the vector field $K^W(\cdot,y)\beta$, the previous equation can be written as

$$\left\langle K^W(\cdot,x)\alpha, w \right\rangle = \alpha^T w(x) \tag{1.9}$$

and since the set of vector fields is dense in the RKHS W, this last equation still holds for any vector field w in W. This is called the "reproducing property" since every vector field "reproduces" itself in the points of the other vector field denoted by the basis element.

This new test space brings several advantages.
First of all, the standard deviation λ^W determines the rate of decay of the RKHS norm to zero. This means that when one is computing the norm of the difference between two different currents (i.e. $\|$T-T'$\|_{W*}$, where W* is the dual space of W, which means the space of currents) this parameter sets the "distance" within which the two currents start to interact. Or, in other words, the shape dissimilarities are captured up to its magnitude [Dur10]. So if this parameter is set to zero, then the RKHS norm tends to the mass-norm. While when this parameter is too big, the vector field is too much smoothed and it can not adapt to the small-scale variations of the two currents (curves or surfaces).
Second, with this new test space, the norm defined by a supremum has a closed form, making the overall framework particularly well suited from a computational point of view (See later Eq. 1.15).

Also in this new context, it will be denoted as W* the dual space of the RKHS W, which means the space of *currents*. One of the key property of the RKHS W is that there is a canonical isometric map between W and its dual space W*. This mapping is denoted L_W, it maps every vector field $w \in$ W to L_W(w) $\in$ W* and it is defined by:

$$L_W(w)(w') = \langle w, w' \rangle_W \tag{1.10}$$

$\forall$ w,w'$\in$ W. Thus L_W(w) is a current (i.e. a mapping from W to $\mathbf{R}$). In this context the dual representation of the basis vectors $K^W(\cdot,x)\alpha$ are exactly the Delta Dirac currents: $\delta_x^\alpha = L_W(K^W(\cdot,x)\alpha)$. Combining Eq. 1.10 and Eq. 1.9 with this last one, one obtains:

$$\delta_x^\alpha(w) = L_W(K^W(\cdot,x)\alpha)(w) = \left\langle K^W(\cdot,x)\alpha, w \right\rangle_W = \alpha^T w(x) \tag{1.11}$$

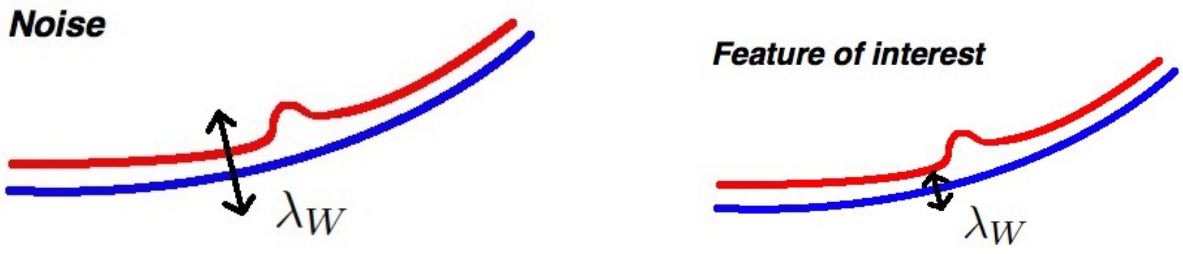
which is another definition of Delta Currents but with RKHS as test space.

Therefore, with this new test space, being W* a vector space of linear maps, it's also provided with the supremum operator norm which is equal to $\|T\|_{W*} = \sup_{\|w\|_W \leq 1} |T(w)|$. Thus, using the isometric map L_W, the norm of the difference between two currents can be written as:

$$\|T - T'\|_{W*} = \sup_{\|w\|_W \leq 1} |T(w) - T'(w)| = \|L_W^{-1}(T - T')\|_W \tag{1.12}$$

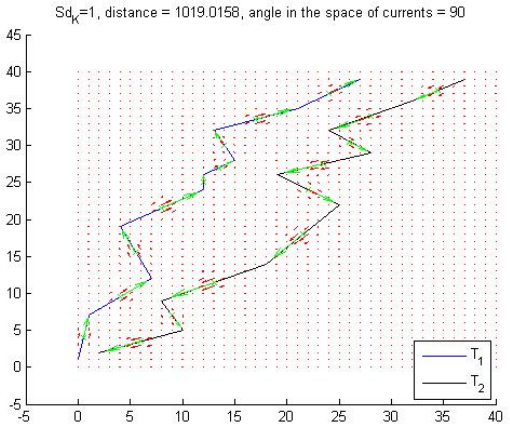
and the supremum is achieved for a particular vector field denoted by $w = L_W^{-1}(T - T')$. This means that if T and T' are two curves, this vector field is the one which maximizes the difference between the integrals $T(w) = \int_T w(x)^T \tau(x) dx$ and $T'(w) = \int_{T'} w(x)^T \tau'(x) dx$ over all possible vector fields $w \in$ W. In some sense, this vector field w is the one which best separates the two curves [Dur10].

Of course, the result of the norm 1.12 depends on the choice of the test space W, which is uniquely linked to the Kernel used, and in this project (since we are going to use only Gaussian Kernels) to the value of the standard deviation λ_W. Thus, as it is possible to notice from Fig.1.4, changing λ_W one changes also the resulting vector field. The smaller the magnitude of λ_W, the higher the spatial frequencies not smoothed of w, and consequently the maximizing vector field w captures more differences between the two curves T and T'.
λ_W is thus chosen in relation to the scale of detail that one wants to have in comparing two different curves or surfaces. Practically λ_W determines the scale under which shape variations are considered as noise. Looking at Fig.1.3 one can notice that the small bump in the red curve is not seen by the algorithm in the left figure since λ_W is too big and the bump is smoothed while it is seen as a shape dissimilarity in the right figure since λ_W is smaller than the distance between the two curves red and blue.

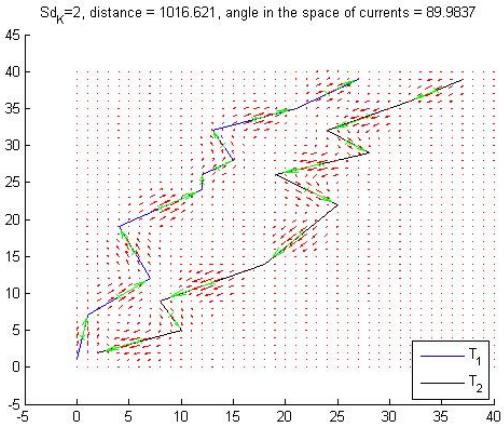


(a) Noise **(b)** Feature of interest

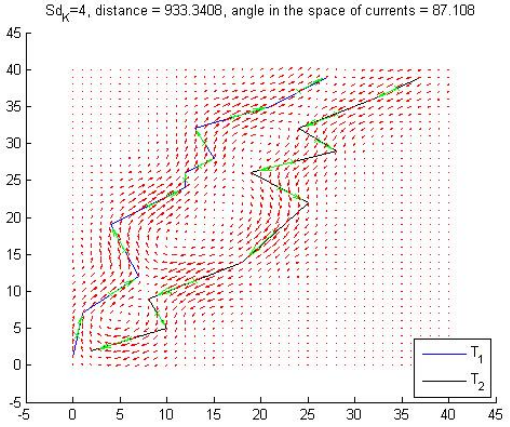
Figure 1.3: Impact of the kernel std. on the distance between two curves T and T'. Picture taken from [Dur10].



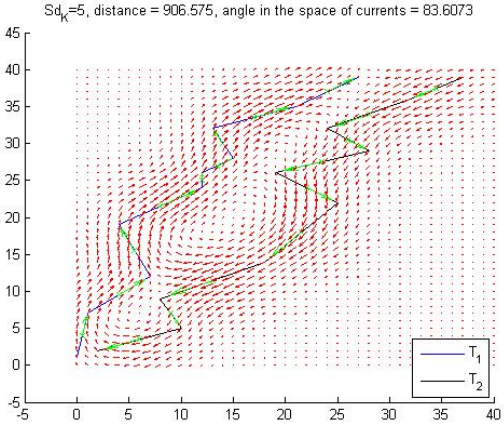
(a) λ_W=1



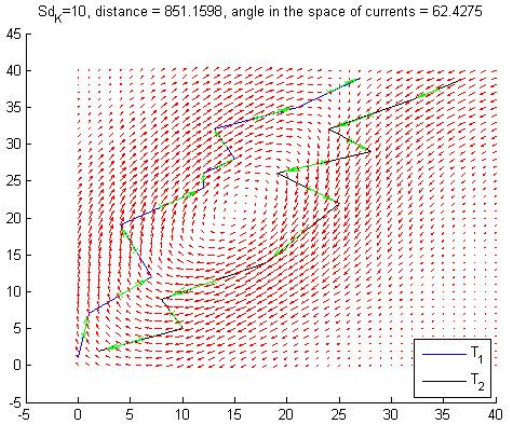
(b) λ_W=2



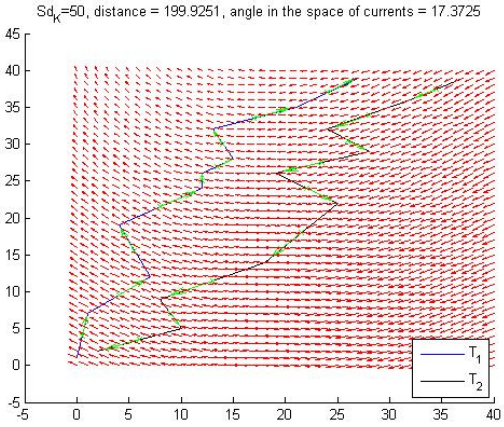
(c) λ_W=4



(d) λ_W=5



(e) λ_W=10



(f) λ_W=50

Figure 1.4: Impact of the standard deviation of the Kernel on the resulting vector field of the norm of the distance between two curves T and T'.

The results shown in Fig.1.4 highlights the effect of the kernel size (dependent on λ_W). For small values of λ_W the vector field can vary fast enough to follow almost every change in direction of both curves (see Fig.1.4b) but only when the size of the kernel is close to the distance between the two curves, the vector field can almost perfectly interpolate all the points between the directions of the curves [Dur10] (see Fig.1.4c and Fig.1.4d). In the first case, with λ_W=2, the two curves are almost orthogonal in the space of currents with an angle equal to $\arccos\left(\frac{|\langle T_1,T_2\rangle_{W*}|}{\|T_1\|_{W*}\|T_2\|_{W*}}\right) = 89.98^o$. Instead, for large values of λ_W (see Fig.1.4e, Fig.1.4f) the vector field can't vary fast enough to follow all the small variations in the directions of the two curves since the highest spatial frequencies are smoothed out. Consequently the two curves become more and more aligned in the space of currents (the angle with λ_W=50 is equal to 17.4^o) and thus the value of the inner product decreases and as a consequence one obtains a smaller value of the distance between the two curves bigger is the value of λ_W.

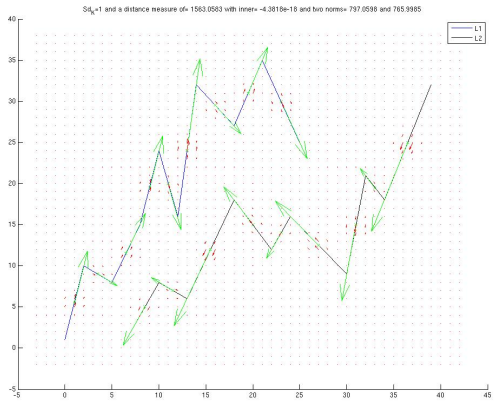
One of the consequences of this behaviour is shown in Fig.1.5a and Fig.1.5b where when λ_W is too small the norm of the difference between the two curves remains the same also translating one of the curve in another position. In fact, the inner product is always equal to zero. This means that, as shown before, the behaviour of this norm is similar to the one of the mass norm and therefore it is useless unless the two curves don't superimpose. Instead, when the value of λ_W is close to the average distance between the two curves, the vector field can interpolate almost perfectly all the points in between them and this means that the norm is no more blind to the positions of the two curves. In fact, as it is possible to see comparing Fig.1.5c and Fig.1.5d, the norm of the difference between the two curves changes translating one of the curves.

Since the norm in the space of currents (Eq.1.12) derives from an inner product, one can rewrite it using the following usual formula:

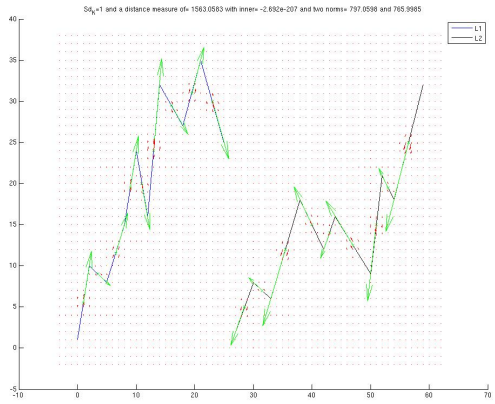
$$\|T - T'\|^2_{W*} = \|T\|^2_{W*} + \|T'\|^2_{W*} - 2\langle T, T'\rangle_{W*} \tag{1.13}$$

Using the second property of the RKHS shown before (Eq.1.8) and the fact that there is an isometric mapping between W and its dual space W*, it's possible to compute the inner product between two Dirac delta currents in the following way:

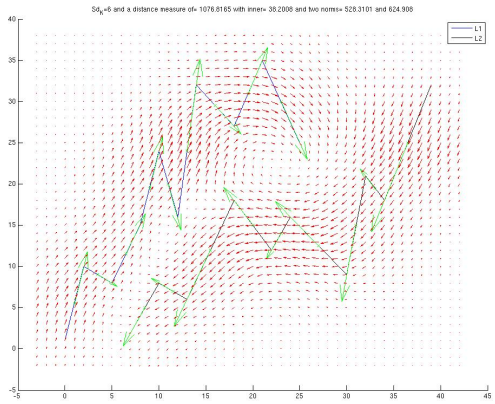
$$\left\langle \delta_x^\alpha, \delta_y^\beta \right\rangle_{W*} = \left\langle K^W(\cdot,x)\alpha, K^W(\cdot,y)\beta \right\rangle_W = \alpha^T K^W(x,y)\beta \tag{1.14}$$



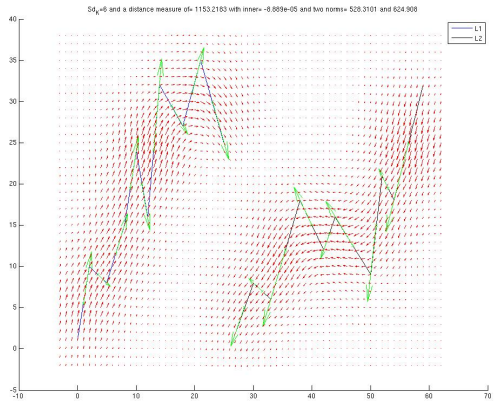
(a) λ_W=1, norm difference = 1563



(b) λ_W=1, second curve translated, norm difference = 1563



(c) λ_W=6, norm difference = 1076



(d) λ_W=6, second curve translated, norm difference = 1153

Figure 1.5: Effect of the translation of one curve on the norm of the difference changing the value of λ_W.

and thus, by linearity, the inner product between two finite sets of Dirac delta currents $T = \sum_{i=1}^{N} \delta_{x_i}^{\alpha_i}$ and $T' = \sum_{i=1}^{N'} \delta_{y_i}^{\beta_i}$ is given by:

$$\langle T, T' \rangle_{W*} = \left\langle \sum_{i=1}^{N} K^W(\cdot, x_i)\alpha_i, \sum_{i=1}^{N'} K^W(\cdot, y_i)\beta_i \right\rangle_W = \sum_{i=1}^{N}\sum_{j=1}^{N'} \alpha_i^T K^W(x_i, y_j)\beta_j \qquad (1.15)$$

This equation gives *explicit* and *easy tractable closed form* formula to compute the inner product between two discrete shapes [Dur10].

Thus the norm of the difference between two curves T and T' can be computed in the following way:

$$\|T-T'\|_W^2 = \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i^T K^W(x_i,x_j)\alpha_j + \sum_{i=1}^{N'}\sum_{j=1}^{N'}\beta_i^T K^W(y_i,y_j)\beta_j - 2\sum_{i=1}^{N}\sum_{j=1}^{N'}\alpha_i^T K^W(x_i,y_j)\beta_j \tag{1.16}$$

Using Eq.1.10 and the fact that L_W is an isometric map, one can also rewrite the norm of the difference between two curves in a different way:

$$\begin{aligned}\|T-T'\|_{W*}^2 &= \langle T-T', T-T'\rangle_{W*} = \left\langle L_W^{-1}(T-T'), L_W^{-1}(T-T')\right\rangle_W \\ &= L_W\left(L_W^{-1}(T-T')\right)\left(L_W^{-1}(T-T')\right) = (T-T')\left(L_W^{-1}(T-T')\right)\end{aligned}$$

and calling the vector field $\left(L_W^{-1}(T-T')\right) = w$, it results $(T-T')(w) = T(w) - T'(w)$. Thus, using the approximation seen before in Eq.1.3, both T and T' can be discretized as finite sets of momenta, $(\{x_i,\alpha_i\})$ for T and $(\{y_j,\beta_j\})$ for T', resulting in:

$$\|T-T'\|_{W*}^2 = T(w) - T'(w) = \sum_{i=1}^{N} w(x_i)^T\alpha_i - \sum_{j=1}^{N'} w(y_i)^T\beta_j \tag{1.17}$$

The vector field $\left(L_W^{-1}(T-T')\right) = w$, belonging to the RKHS, is equal to the convolution of the momenta (x_i,α_i) and (y_j,β_j) by the kernel K^W, namely:

$$w(x) = L_W^{-1}(T-T')(x) = \sum_{i=1}^{N} K^W(x,x_i)\alpha_i - \sum_{j=1}^{N'} K^W(x,y_j)\beta_j \tag{1.18}$$

Combining these two last Equations 1.17 and 1.18 it results exactly Eq.1.16.

1.3.1 An efficient computational framework

As seen in the previous section, it exists an easy way to compute the norm of a current via a double sum: $\|T\|_{W*}^2 = \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i^T K^W(x_i,x_j)\alpha_j$ which can be rewritten in matrix notation as $\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha}$.

This offers an efficient way to compute the norm of a current via matrix multiplication when the number of momenta is finite and not too big. Unfortunately in this project the size of meshes and curves is such that it is necessary to find a more efficient way to compute the metric on currents.

In fact, the main problem is that it is necessary to compute a different matrix $\mathbf{K}$ for every current, since it depends on the points of the curves or surfaces. Thus a possible solution is to introduce a linearly spaced grid which defines a finite-dimensional sub-space of the RKHS W and therefore also a sub-space of the space of currents W* [Dur10]. In this way, momenta will be constrained to be located only in a particular set of points Λ and the matrix $\mathbf{K}$ can be pre-computed and stored once for all, at the beginning of the process, allowing also a fast parallelization of the computation on currents.

It was shown in [Dur10] that when the set Λ becomes dense in $\mathbf{R}^d$, the approximation space converges to the original space. Therefore, all the results and equations shown so far can be brought also in this new context. So the new subset of the RKHS W will be called W_Λ and a basis of this vector space will be given by the vector $K(x,\cdot)\epsilon_k$ for any point x belonging to the set Λ ($\in \mathbf{R}^d$) and for any ϵ_k being an orthonormal basis of $\mathbf{R}^p$[2]. So, if the number of points in W_Λ is N, its dimension is Np since all the points belong to $\mathbf{R}^p$ and any vector field $\gamma \in W_\Lambda$ has a unique decomposition of the form:

$$\gamma(x) = \sum_{i\in\Lambda} K(x, x_i)\alpha_i \tag{1.19}$$

This last equation is exactly the one seen before as first property of the RKHS but with one important difference. Now this decomposition is unique since the set of points (and therefore momenta) is constrained and not arbitrary and so the dimension of W_Λ is equal to Np and not infinite. In fact, by contrast to infinite dimensional spaces, a set of linearly independent vectors which spans a finite dimensional space is a basis of the vector space. Therefore the extra condition needed before in the infinite dimensional space, like the orthonormal condition, is no more necessary. It's important to enhance the fact that Dirac Delta Currents are linearly independent and span the space of currents but they are not orthogonal ($\left\langle \delta_x^\alpha, \delta_y^\beta \right\rangle_{W*} = \alpha^T K^W(x,y)\beta \neq 0$) and that's why the decomposition of a current as linear combination of Dirac Delta Currents in the previous infinite space W was not unique.

Also in this case we can rewrite all the equations seen before in matrix notation: calling $\boldsymbol{\alpha}$ the concatenation of the N vectors α_i of dimension p and $\boldsymbol{\gamma}$ the concatenation of the N vector fields $L_W^{-1}(T)(x_i)$ of dimension p, one can write $\boldsymbol{\gamma}=\boldsymbol{K}_\Lambda\boldsymbol{\alpha}$, where $\boldsymbol{K}_\Lambda$ is the Np-by-Np block-circulant matrix. From a computational point of view, the norm and inner product

[2]In this project $\mathbf{R}^d$ is always the set for points x or y while $\mathbf{R}^p$ is the set for vectors like α or β. From a practical point of view both d and p are always equal to 3.

operations have a complexity equal to N^2p^2 which is the number of elements of $\boldsymbol{K}_\Lambda$. But, since this matrix is built on a linearly spaced grid with periodic boundary conditions [3] and with blocks given by scalar and translation-invariant kernels (i.e. K(x,y)=K(|x-y|)Id), it can be reduced to a circulant matrix with N^2 elements. An example of block-circulant matrix is:

$$\begin{bmatrix} K_0 & K_1 & K_2 & ... & K_N \\ K_N & K_0 & K_1 & ... & K_{N-1} \\ ... & ... & ... & ... & ... \\ K_1 & K_2 & K_3 & ... & K_0 \end{bmatrix}$$

In fact, using as example p=2 (which means a 2D representation), each block of the $\boldsymbol{K}_\Lambda$ is equal to $K = \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix}$ where b is equal to $\exp(-[(x_i - x_j)^T(x_i - x_j)]/\lambda^2)$ (i.e. Gaussian Kernels). Moreover, being the grid linearly spaced with periodic boundary conditions, $K(x_i, x_j) = K(x_0, x_{j-i})$ for $0 \leq i, j \leq N-1$, which means that $\boldsymbol{K}_\Lambda$ is indeed a block-circulant matrix where each block is a matrix K(x,y). But since every matrix K(x,y) has only one value different from zero, it can be taken into account only this value b in the computation, reducing the number of elements from N^2p^2 to N^2.

Therefore, the matrix $\boldsymbol{K}_\Lambda$ becomes a circulant matrix with N^2 elements and it's completely determined by its first row $\mathbf{k}$ of size N.

So any vector field $\boldsymbol{\gamma} = L_W^{-1}(T)$ associated to a generic current T, restricted to the set of momenta given by Λ, can be computed with p discrete convolutions between the first row $\boldsymbol{k}$ of the matrix $\boldsymbol{K}_\Lambda$ and $\boldsymbol{\alpha}$ which has been divided in p vectors of size N:

$$\boldsymbol{\gamma}_i = (\boldsymbol{K}_\Lambda \boldsymbol{\alpha})_i = \boldsymbol{k} \star \boldsymbol{\alpha}_i \quad 1 \leq i \leq p \tag{1.20}$$

where $\star$ denotes the discrete convolution between matrices.

Each discrete convolution between matrices can be computed via FFT at a cost $O(N\log(N))$: $A \star B = FFT^{-1}(FFT(A) \cdot FFT(B))$, where $\cdot$ denotes the element-wise multiplication of matrices. Therefore the total cost is $O(pN\log(N))$ which is definitely lower than the direct computation using the double sum $O(N^2p^2)$. [4]

Due to the periodic boundary conditions of the grid, the result of the computation in Eq. 1.20 using FFT can also be not strictly equal to the result using discrete convolution. This

[3] It means that the grid has the geometry of a torus: points at two opposite borders are at distance 0 from each other.

[4] The number of points in the grid is chosen as a power of two ($\Lambda = 2^M$) for FFT computation purpose.

is because momenta close to a border of the grid may influence the result of the vector field at a point close to the opposite border. In order to avoid that, the grid borders are enlarged and located at a distance equal to at least $3\lambda_W$ from every momenta. It's important to recall that we are using Gaussian Kernels.

The next step is how to project a current T onto the grid W_Λ. As shown in [Dur10] the best approximation (in the sense of the norm in W) is given by the orthogonal projection of the complete vector field $\gamma \in$ W on W_Λ and consequently of T on W^*_Λ (dual space of W_Λ). Unfortunately it turned out that this operation is ill-posed from a numerical point of view (see [Dur10] for more information). Therefore another strategy is used: Partial Volume Projection (PVP) with which all the components of the momenta are projected in different grids. Using as example a 3D representation, every momenta has three components (x,y,z) and for each one of them a grid is created. These grids are divided in "cubes" where the components are first inserted and then projected at the surrounding 8 nodes of the "cube" where they belong to. In fact, one can split a 3D momenta (a point joined to a vector: an oriented point) into three different "weighted-points" which are defined as the points in 3D linked to each component of the vector: ($[\boldsymbol{x}, \alpha_x]$,$[\boldsymbol{x}, \alpha_y]$,$[\boldsymbol{x}, \alpha_z]$). Each one of these "weighted-points" is afterwards projected to a linearly spaced grid (seen before) with the technique of the PVP:

1. Insert the point $\boldsymbol{x}$ in the grid
2. Find the 8 nodes around $\boldsymbol{x}$
3. Add to each node the value of the component under examination α_i multiplied by an interpolating functions ρ_i

It's useful to recall the definition of interpolating function ρ_i:

- $\sum_i \rho_i(x) = 1$
- $\sum_i \rho_i(x)(x\text{-}x_i)=0$

In three dimensions, each grid is divided into cubes. A point inside a cube creates 8 sub-volumes. $\rho_i(x)$, where x is the point inside the cube and i the index related to one of the 8 nodes, is equal to the sub-volume opposite to the node i divided by the length of the cube (l) to the power of 3 (l^3).

At this point, one has 3 linearly spaced grids with periodic boundary conditions (one for each component x,y,z), and using the FFTs as seen before it can be computed every component of the vector field at each point giving as result a 4D matrix (3 grids of 3D).

The FFT of the Kernel is computed once for all at the beginning of the process since the three grids are exactly the same and therefore also the distances between grid points (the Kernel depends only on the distance between two poins since it's Gaussian). After that the FFT of the α_i values projected at each grid point is computed (for each grid). The inverse FFT applied to the product of the two FFTs (Kernel and α_i) for each grid gives as result the vector field. Now, computing the sum of every point value squared at the same position in each grid, one obtains exactly the "total" vector field at each grid point. But if one wants to compute the vector field in another series of points (not at the grid points), it is necessary to have an interpolation scheme.

The third step is therefore to define an interpolation scheme which can reconstruct a dense vector field from samples on grid nodes. Or, in other words, to find the values of the vector field in a series of points different from the grid nodes. A natural choice of interpolation scheme is the PVI (Partial Volume Interpolation) since it was used before PVP. It is exactly the inverse operation. In fact, each new point y can be associated to a voxel of each grid (the cube whose nodes are around y) and every component of the vector field can be computed using the following equation:

$$f(y)_j = \sum_i \rho_{i,j}(y) f_{i,j} \tag{1.21}$$

where j is the component (x,y,z) of the vector field and i is the index for the nodes. Thus the result of this operation in a space $\mathbf{R}^3$ is a series of vectors $\in \mathbf{R}^3$, one for each new point y.

All the demonstrations about the construction of a finite-dimensional approximation of RKHS and how it converges to the original space can be found in [Dur10]. Moreover it's also interesting to remark that always in [Dur10] it was demonstrated that in this framework the approximation error depends only on the ratio between the grid step Δ and the rate of decay of the kernel λ_W, which is set equal to 0.2 in this project.

1.4 Sparse Representation of currents

As seen before, a surface or a curve can be approximated by a finite sum of Dirac Delta Currents. This approximation may be redundant at the scale of analysis which is given by the standard deviation of the Kernel λ_W. This means that two segments, approximated by two Dirac Delta Currents which are closer than λ_W, are almost undistinguishable in the space of currents. This results from the effect of the smoothing Kernel. In fact, the action of any vector field w on the current $\delta_x^\alpha + \delta_y^\beta$ gives as result $w(x)^T\alpha + w(y)^T\beta =$

$K(\cdot,x)\alpha + K(\cdot,y)\beta$ and if $|x-y| \ll \lambda_W$, the vector field is almost constant between x and y and therefore $w(x)^T\alpha + w(y)^T\beta$ might be well approximated by $w(\frac{x+y}{2})^T(\alpha+\beta)$. This means that the sum of the two initial Dirac Delta Currents $\delta_x^\alpha + \delta_y^\beta$ may be approximated by a single one $\delta_{\frac{x+y}{2}}^{\alpha+\beta}$.

Exploiting this idea, it's possible to approximate the momenta located in every patch of size λ_W by a single momenta and therefore to have a sparser representation. This means also that the resulting Dirac Delta Currents are almost orthogonal since their relative distances are greater than λ_W, which means that taking any couple of resulting momenta (x,α) and (y,β) their inner product will be close to zero ($\left\langle \delta_x^\alpha, \delta_y^\beta \right\rangle_{W*} \sim 0$) since we are using Gaussian Kernels ($\exp(-|x-y|^2/\lambda_W^2) \sim 0$ if $|x-y| \gg \lambda_W$).

In an optimal decomposition one would like also to choose the momenta in order that the new approximation is similar to the original current. Or, in other words, one looks for the set of Dirac Delta Currents ($\sum_{i=1}^N \delta_{x_i}^{\alpha_i}$) that have a strong correlation with the original current T, which means maximizing $\left\langle \sum_{i=1}^N \delta_{x_i}^{\alpha_i}, T \right\rangle_{W*}$.

A greedy approach to this idea is the one introduced in [Dur10]: a matching pursuit algorithm. Matching pursuit algorithm is based on the iteratively research of the linear combination of atoms (momenta in this project) that best correlates with the residual function. Of course, at the beginning of the process, the residuals are equal to 0 and therefore one looks for the atom that best correlates with the initial function. The basic idea is therefore to reduce an initial dictionary of atoms to a sparser one and the resulting subset is the one which can best explain the initial function. In the context of currents, given an initial current $T = \sum_{i=1}^M \delta_{x_i}^{\alpha_i}$, one looks for the set of momenta $\{\sum_{i=1}^N \delta_{y_i}^{\beta_i}\}$ that best approximates T (with $N \ll M$).

If one knows the positions of the points of the best set of momenta $Y_n = \{y_i\}_{1\leq i\leq N}$, it can be defined a sub-space of currents, as in the previous section, called $W_{Y_n}^*$,where the previous subset Λ is substituted with Y_n for a clarity purpose. This subset is the space given by all the linear combinations of the momenta $\delta_{y_i}^{\epsilon_k}$, with y_i being the points belonging to Y_n, $k = 1,2,3$ since we are dealing only with curves and surfaces in 3D and ϵ_k is a canonical basis of R^3 (i.e. $W_{Y_n}^* = \text{span}(\delta_{y_i}^{\epsilon_k})$). After that, one needs to find the vector β_i linked to each best point inside this new sub-space, and the closest approximation to T in $W_{Y_n}^*$ is given by the orthogonal projection of T onto $W_{Y_n}^*$, which means $\left\langle T, \delta_{y_i}^{\epsilon_k} \right\rangle_{W*} = \left\langle \sum_{p=1}^N \delta_{y_p}^{\beta_p}, \delta_{y_i}^{\epsilon_k} \right\rangle_{W*}$. Both parts of this last equation can be rewritten in the following way:

$$\begin{aligned}\left\langle T, \delta_{y_i}^{\epsilon_k} \right\rangle_{W*} &= \left\langle L_W^{-1}(T), L_W^{-1}\left(\delta_{y_i}^{\epsilon_k}\right) \right\rangle_W \\ &= \langle \gamma, K^W(\cdot, y_i)\epsilon_k \rangle_W \\ &= \gamma(y_i)^T \epsilon_k \\ &= \gamma(y_i)_k\end{aligned} \tag{1.22}$$

where $\gamma = L_W^{-1}(T)$ and it was used the definition of Dirac Delta Currents: $\delta_x^\alpha = L_W(K^W(\cdot, x)\alpha)$, the reproducing property from Eq1.9 and the isometric mapping L_W.

$$\begin{aligned}\left\langle \sum_{p=1}^N \delta_{y_p}^{\beta_p}, \delta_{y_i}^{\epsilon_k} \right\rangle_{W*} &= \left\langle L_W^{-1}\left(\sum_{p=1}^N \delta_{y_p}^{\beta_p}\right), L_W^{-1}\left(\delta_{y_i}^{\epsilon_k}\right) \right\rangle_W \\ &= \sum_{p=1}^N \epsilon_k^T K^W(y_i, y_p)\beta_p \\ &= \sum_{p=1}^N \left(K^W(y_i, y_p)\beta_p\right)_k\end{aligned} \tag{1.23}$$

where it was used Eq.1.14. Merging these two equations, one obtains a set of 3N linear equations:

$$\sum_{p=1}^N \left(K^W(y_i, y_p)\beta_p\right)_k = \gamma(y_i)_k \tag{1.24}$$

solving this linear system leads to the values of the N optimal vector β_i linked to the best positions Y_n.

Finding the optimal points $\{y_i\}_{1 \le i \le N}$ has been proved to be NP-hard in general [Dur10] [DMA97]. The orthogonal matching pursuit algorithm is a suboptimal greedy approach to this problem. At each iteration one chooses as new optimal point the one for which the projection of the residual current onto $W_{Y_n}^*$ is maximal. The first point is chosen using the whole initial current T. Thus:

$$\langle T, \delta_z^{\epsilon_k} \rangle_{W*} = \langle \gamma, K^W(\cdot, z)\epsilon_k \rangle_W = \gamma(z)_k \tag{1.25}$$

and y_1 (the first optimal point) is the one for which $|\gamma(z)|$ is maximal. It's important to notice that $\{z\}$ are all the points of the grid where T is projected (see Section before) but they are not the points of the momenta of T, which are called $\{x\}$. On the other hand, $\{y\}$ are the optimal points resulting from the Matching Pursuit algorithm and they are chosen from the whole set of points in the grid $\{z\}$ and, therefore, they can be different from $\{x\}$.

After finding the first optimal point y_1, one solves Eq.1.24 obtaining $\beta_1^1 = K^W(y_1, y_1)^{-1}\gamma(y_1)$, which is the value of β for the first optimal point at the first iteration. After that one removes from the original vector field γ its orthogonal projection on ${L_W}^{-1}(\delta_{y_1}^{\epsilon_k})$ for each dimension k, which gives the first residual vector $\gamma_1(z) = \gamma(z) - K^W(z, y_1)\beta_1^1$. It should be clear that only the $\{z\}$ points closer than $3\lambda_W$ from y_1 are modified in a significant way by this operation since we are using a Gaussian Kernel with standard deviation equal to λ_W.

After n steps the algorithm finds y_n which is the point where the residual vector field γ_{n-1} is maximal. Solving Eq.1.24, $\sum_{p=1}^{n}\left(K^W(y_n, y_p)\beta_p\right)_k = \gamma_{n-1}(y_n)_k$, leads now to n vectors β: $\beta_1^n, \beta_2^n, ..., \beta_n^n$ and therefore the approximation of T on $W_{Y_n}^*$ becomes: $\sum_{i=1}^{n}\delta_{y_i}^{\beta_i^n}$.

Therefore at each step one finds an optimal point and removes the "effect" of this point and of all the other optimal points from the original vector field. The magnitude of the "effect" of the optimal points is given by the standard deviation of the Kernel λ_W. And it's exactly like removing at each step a "patch" centred on each optimal point whose size is defined by λ_W.

It was proven in [Dur10] that the approximation $\sum_{i=1}^{n}\delta_{y_i}^{\beta_i^n}$ converges towards T as n tends to infinity, which means $\|T - \sum_{i=1}^{n}\delta_{y_i}^{\beta_i^n}\|_{W*} \to 0$ when $n \to \infty$. Moreover it was also demonstrated that the L^∞-norm of the residual vector field tends towards zero as n tends to infinity ($\|L_W^{-1}(T) - L_W^{-1}\left(\sum_{i=1}^{n}\delta_{y_i}^{\beta_i^n}\right)\|_\infty \to 0$, when $n \to \infty$). This means that one can use the infinity norm of the residual vector field as stopping criteria, since it will go below a certain threshold η after a finite time.

Depending on the application, the threshold η can be chosen in different ways. If one is dealing with sets of objects from different subjects, one can choose a certain percentage of the standard deviation as threshold. Let $\{T_i\}$ be a set of objects of the same kind from different subjects and N the total number of subjects, one can write the variance as: $\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}\|T_i - \overline{T}\|_\infty$ where $\overline{T}$ is the mean $\frac{1}{N}\sum_{i=1}^{N}T_i$. And the Matching Pursuit algorithm stops when the residual vector field $\|\gamma_n\|_\infty$ is smaller than a certain percentage τ of the standard deviation. In this way one can assure that the mean of the approximations is not so far from the initial mean since the new currents $\sum_{i=1}^{n}\delta_{y_i}^{\beta_i^n}$ have a distance smaller than $\tau\%$ of the std from the initial T_i.

A sketch of the Matching Pursuit algorithm is presented in Table 1.1.

Algorithm - Orthogonal Matching Pursuit for Currents

Input: vector field γ, standard deviation σ, percentage threshold $\tau>0$

0: Set $\gamma_0 = \gamma$,n=0
1: **while** $\|\gamma_n\|_\infty \geq \tau\sigma$ **do**
2: $y_{n+1} = \text{argmax}_{y\in\mathbf{R}^r}|\gamma_n(y)|$
3: Find $(\beta_i^{n+1})_{1\leq i\leq n+1}$ by solving $\sum_{q=1}^{n+1}\left(K^W(y_i,y_q)\beta_q^{n+1}\right)_k = \gamma(y_i)_k$
4: $\gamma_{n+1} = \gamma - \sum_{i=1}^{n+1} K^W(\cdot,y_i)\beta_i^{n+1}$
5: $n \leftarrow n+1$
6: **end while**
Output set of momenta $\{y_i,\beta_i^n\}_{1\leq i\leq n}$

Table 1.1: Sketch of the Orthogonal Matching Pursuit Algorithm applied to this project.

Comments about Matching Pursuit representation A sparser representation can be very important for two main reasons: *Compression* and *Interpretation*. Raw anatomical data such as mesh or sets of curves can be composed by many points and therefore it could be very useful to reduce their size in order to speed up any operation conducted with them. Furthermore, using the Matching Pursuit algorithm, one can also reduce the redundancy of information present in the initial current. As seen before, the scale of analysis is given by the standard deviation of the Kernel λ_W and it determines also the "resolution" of the current since it sets the minimum distance between different momenta. In this way one can decide the degree of detail of an image/mesh/curve keeping only the information that permits a better interpretation.

An example of Matching Pursuit Algorithm applied to a fiber bundle (set of curves) is shown in Fig.1.6 using different λ_W but with the same fixed error (τ=0.05). It is easy to notice that: the greater λ_W, the sparser the approximation. It means that increasing λ_W the number of resulting momenta is smaller but their magnitude is bigger. That's because one should imagine a "patch of influence" around each resulting momenta which increases with λ_W. All the segments passing through this patch are approximated by the new resulting momenta. Therefore, more curves intersect this patch, bigger is the magnitude of the momenta.

Looking at Fig.1.6 it seems quite difficult to decide which is the best sparse representation of the initial bundle. This choice depends on the degree of detail one wants to keep. If one is looking for an accurate and precise representation where also the central bending of the bundle is kept, a good choice could be λ_W=1. Instead, if one wants just to restrain the main shape of the bundle a value of 3 or greater would be a better choice.

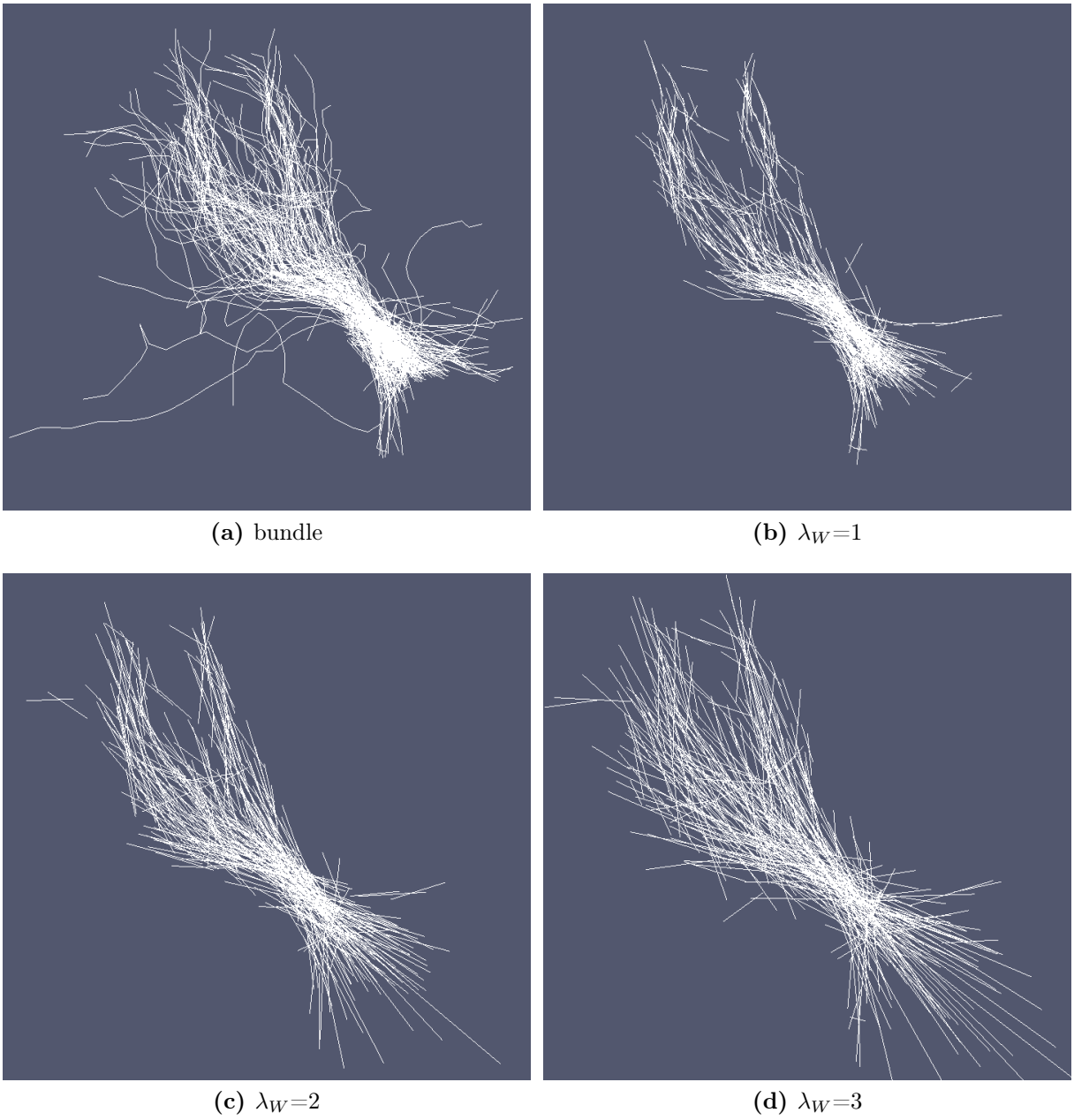
Unfortunately, this kind of sparse representation has also a drawback. It can't keep the connectivity between points. This means that, as it is possible to notice from Fig.1.6, the resulting representation is composed only by momenta (oriented points) and not by segments. Therefore one can't use this sparse representation for other processes in which it's important to keep the information about connectivity. For example, in this project, we want to check which are the parts of the brain, among which also brain bundles, which show a greater variability between a population of healthy controls and of patients subject to Gilles de la Tourette syndrome. In order to do that, we need to keep the connectivity between the points of the fibers. In fact, without this information, it would be impossible to understand exactly which are the parts of the bundle more variable.

Thus, on one hand we would like to have a sparse representation to speed up the computation and, in some cases, also for a better interpretation. But, on the other hand, we need to keep the connectivity of the fiber bundles to comprehend which are the parts of the brain bundles more variable. A possible solution to this problem is the technique of atlas construction (Chapter 3 and Chapter 4) where an average template is estimated using all the objects of the population. The template keeps the connectivity between points while all the other objects are modified using the Matching Pursuit algorithm. After that the template is moulded such that it shows all the common features among the population. Deforming and matching afterwards the template towards each object gives the possibility to understand which are the parts of the template more variable inside the population. This process is repeated for both the two populations (controls and patients) in order to check if the most variable parts in each template are different. Of course this raises also another question about the initialization of the bundle template. A possible solution will be shown in Chapter 3.

In conclusion, it's important to highlight the fact that this technique is not used for surfaces (mesh). In fact, as seen before, this sparse representation looses the initial topology of the object. Therefore, during the matching of the template in the atlas construction, it would be more difficult to superimpose a surface template on a surface object without keeping a closed surface or, in other words, the same topology between them. This can be seen as a constrain that we decided to use in order to be sure that similar parts belonging to the template and to the object match together.

In contrast with that, fiber bundles are the results of tractography algorithms. These algorithms were never shown to produce stable and reproducible results, therefore these bundles shouldn't be compared relying only on individual fibers but rather on the global shape of the bundle. In fact, fiber bundles present a great variability among them and it would be really difficult (probably impossible) to match perfectly two bundles considering also all the complete fibers (keeping the connectivity). It seems a better idea to match a complete template, keeping the connectivity of the fibers, to a set of oriented momenta which is exactly the result of the Matching Pursuit algorithm. In this way we take into account the information about the orientation of the single fibers but at the same time

we can understand also which are the parts of the bundle most variable and above all we can keep the information about the start and end-point of the bundle which are linked with other area of the brain (i.e. cortical surface). Examples of fibers bundles and mesh used in this project will be shown later in Chapter 3.



(a) bundle **(b)** λ_W=1

(c) λ_W=2 **(d)** λ_W=3

Figure 1.6: Matching Pursuit result using different λ_W and the same approximation error (5% of the standard deviation).

1.5 Conclusion about Currents

Currents represent an interesting framework to work with geometrical data. In fact they represent a vector space where it is defined an inner product and therefore also a natural metric. Moreover this is a space of continuous linear mapping from a test space given by the convolutions between any square integrable vector fields and a smoothing kernel (RKHS) to $\mathbf{R}$. This permits also to have an isometric mapping between the vector field space and the space of currents but above all an *explicit* and *easy tractable* formula to compute the inner product between two shapes and therefore a *measure of distance*.

In addition to that, there is no need to assume any point correspondence (or other feature) between the two shapes. This is extremely important when dealing with objects of different size due to a different sampling or simply due to normal anatomical differences.

Another important quality of this framework is mainly linked with the fiber bundles (sets of connected segments, see Fig.1.6). As seen in the previous section fiber bundles should be compared as a whole and not as individual fibers but on the other hand one shouldn't loose the information about the orientation of the fibers. The framework of currents is exactly the answer of all these problems. In fact, it doesn't see the bundle as a set of unconnected points but it takes into account also the local orientation of the momenta.

Furthermore this framework is also robust to fiber connectivity which is fundamental for the template estimation. One of the main contribution of this project is indeed to estimate an atlas (template) for fiber bundles which keeps the connectivity of the fibers and therefore that can show exactly the most variable parts of the objects of a population.

A sparse representation of currents was also presented. This can be used to compress the data in order to reduce the computational time of the process or also to give an easier interpretation of the results. An important characteristic is that it doesn't keep the initial topology of the object. Therefore it can't be used to represent the bundle template since information about the exact location of the parts showing the greatest variability would be lost.

Chapter 2

Gilles de la Tourette Syndrome

In this section it is presented a brief description of Gilles de la Tourette syndrome, which includes also an explanation of the course, prevalence and causes linked to this syndrome (TS). Moreover it is also described a new hypothesis introduced by Yulia Worbe et al. in [WGH+10] which can have clear implications for the current neuroanatomical model of this syndrome. One of the goals of this project is to start the analysis of this hypothesis using the framework of currents on a data set presented in chapter 3.

2.1 Clinical Description

Tourette sydrome (TS) is a childhood onset inherited neuropsychiatric disorder characterized by the presence of sudden, rapid, recurrent and nonrhythmic movements (motor tics) or sounds (vocal or phonic tics) which are often accompanied by psychiatric comorbidities such as obsessive-compulsive disorder (OCD), attention deficit-hyperactivity disorder (ADHD) or other behavioural problems [Kim10], [MM11] [TM09], [Sin05]. Gilles de la Tourette's name was given to this syndrome by Jean-Martin Charcot, once director of the well known Pitié-Salpêtrière Hospital in Paris, after that his student Gilles de la Tourette described 9 cases with a bizarre neurological condition that he referred to as "maladie de tics" [TM09], [MM11], [dlT85]. The generally accepted research definition of this syndrome is based on four diagnostic criteria:

1. Multiple motor tics and at least one vocal tic have been present at some time during the illness although not necessarily concurrently.

2. Tics occur many times a day, nearly every day throughout a period of more than one year. During this period there must not be any tic-free period of more than three consecutive months.

3. The onset is before age 18 years.

4. Disturbances are not due to any direct physiological effects, drugs or any other general medical condition (i.e. Huntington's disease). [Ass00]

2.2 Course

Typically the disorders begin in early childhood, on average between the ages of 3 and 8 years [Lec02] with transient bouts of simple motor tics such as eye blinking or other facial tics. Phonic tics, such as repetitive bouts of sniffing or throat clearing can begin as early as 3 years of age, but typically they follow the onset of motor tics by several years [Lec02]. These tics may initially wax and wane but eventually they become persistent reaching the maximum tic severity at an age around 8-12 years. After that there is usually a steady decline in symptoms [LZV+98]. However, the most severe cases of TS arise in adulthood with extreme form of self-injurious motor tics or socially unacceptable obscenities and gestures such as *coprolalia* (uttering of obscene words) which is probably the most known effect of TS but it occurs only in about 10% of patients [Lec02].

2.3 Prevalence

Once thought to be rare, current estimates change this view. From 1988 to 1998 it passed from an estimation of 2.9 per 10.000 to 299 per 10.000. In the past ten years the prevalence increased again since it was shown in 2003 that in a cohort of children aged 7-15 years the percentage with TS was 5.6 per 1000 while another study in 2006 showed a result of 3.3 per 1000 in pupils aged 6-12 years [Lec02]. Another work [SBR05] showed that 1% of the schoolchildren in the UK between the ages of 5 and 17 years presented the TS syndrome. All this data show that this syndrome is not as rare as it was thought and an increasing level of interest in the world of research has accompanied these discoveries. In fact, as shown in [MM11], the number of publications about TS in PubMed changed from around 20 in 1970 to 180 in 2008.

2.4 Causes

The causes of this syndrome are still unclear and the tremendous variability of disorders linked to TS suggests that these conditions have probably several different causes. Different living environment and lifestyle factors such as tobacco, stress, infection or autoimmune dysfunction are implicated in the development of Tourette syndrome [MM11]. *Genetic factors* have a major role in the development of TS. Family studies showed that the incidence of TS is 10-100 times greater in first-degree relatives of a patient with TS than in the general population and twin studies have demonstrated higher concordance rates of TS in monozygotic than in dizygotic (50-77 % versus 10-23 %) [MM11]. But with the arrival of new Imaging methods in the last years there has been an increasing interest for the neuroanatomical and neurobiological causes of this syndrome. Many researchers focused their attention on multisynaptic neural circuits (loops) that link the cerebral cortex with subcortical regions. Structural MRI studies have identified regional abnormalities in grey matter especially in the basal ganglia (a group of nuclei strongly connected with cerebral cortex and thalamus such as putamen and caudate) [HSL+10]. All these results support the hypothesis that Gilles de la Tourette syndrome is a neurodevelopmental disorder associated with dysfunction of cortico-striato-thalamo-cortical loops [WGH+10].

2.5 Circuits linking Basal Ganglia and Cortex

From earlier data it had appeared that the basal ganglia served primarily to integrate diverse inputs from the entire cerebral cortex and to "funnel" these influences, via the thalamus, to the motor cortex. In particular, the basal ganglia was thought to provide a route for the information coming from the cortical association areas towards the motor cortex area, participating in the initiation and control of the movement. As demonstrated in [ADS86] cortical projections to the basal ganglia are functionally and topographically organized, leading to the concept of functional divisions of cortico-striato-thalamo-cortical loops into sensorimotor, associative and limbic circuits that are implicated in motor, cognitive and motivational aspects of behaviour respectively (see Fig. 2.2) [WGH+10]. Each one of these circuits receives multiple inputs from several functionally related cortical areas (A,B,C in Fig. 2.1). These projections are partially overlapping and are sent to a restricted portion of the striatum (caudate, nucleus and putamen). These striatal regions send further projections to the globus pallidus and substantia nigra, which in turn project to a specific region of the thalamus. Each thalamic region projects back to one (or more) of the same cortical areas that feeds the circuit at the beginning completing the “closed loop”. As enhanced by Alexander et al. in [ADS86] the term "circuit" is not intended to imply a rigidly self-enclosed pathway without inputs or outputs to other structures. Rather a main and central path in the closed loops that starts from particular and functionally

related areas in the cortex and it terminates in one of them (or more).

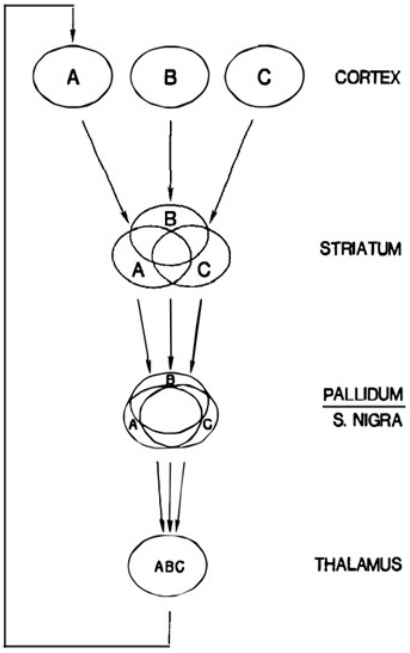


Figure 2.1: Generalized cortico-striato-thalamo-cortical circuit. Image taken from [ADS86]

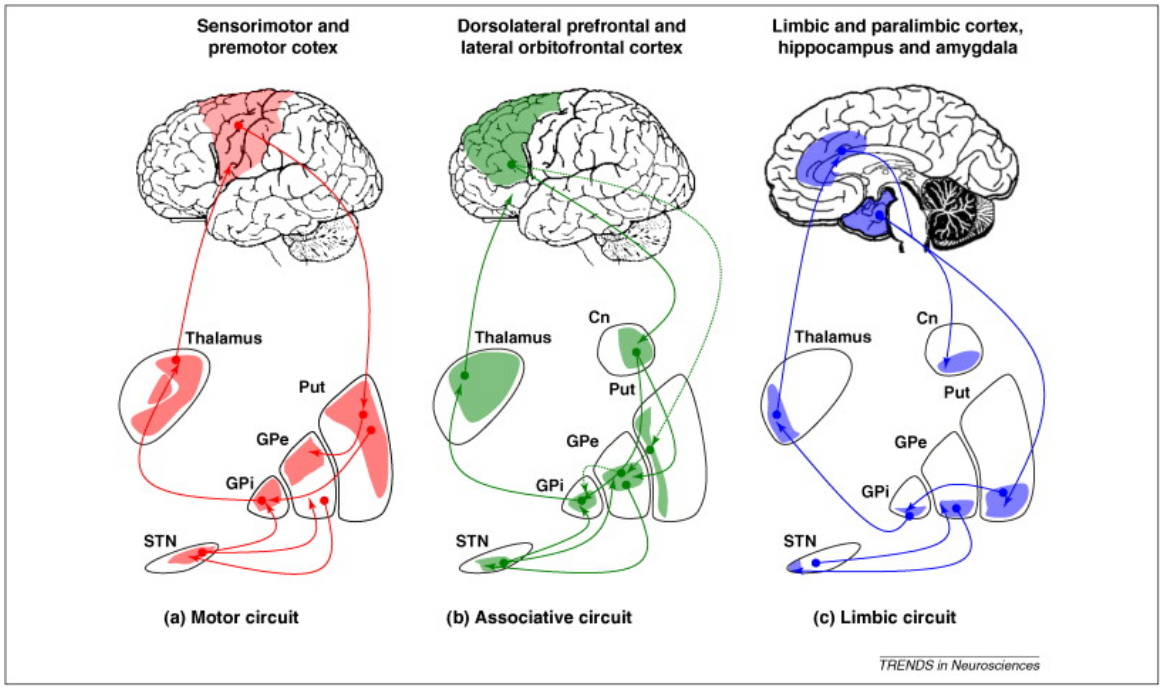


Figure 2.2: Schematic diagram illustrating the main cortico-striato-thalamo-cortical circuits. Image taken from [KHB+10].

2.6 Hypothesis to test

The three most important circuits linked with Tourette's syndrome are the ones presented in Fig. 2.2. According to the model presented before, it was suggested in [Sin05] that the motor tics may result from the dysfunction of premotor and motor circuits, while behavioural disorders may be linked to dysfunction of associative and limbic circuits. This hypothesis was supported by many experiments both in primate model and in humans [WGH+10]. Moreover structural changes were observed in different regions of the cortex and, as said before, also in the striatum, globus pallidus and in the thalamus. Tics and psychiatric co-morbidities have been correlated with cortical thinning in the sensorimotor cortex, with volume reduction in areas such as putamen or caudate nucleus and also with volume increase in the amygdala. Thus, all these results bring evidence of a link between abnormalities in the cortico-striato-thalamo-cortical circuits and the presence of Tourette's syndrome.

Eventually Worbe et al. [WGH+10] divided a set of patients in 3 different groups:

1. simple tics
2. simple and complex tics
3. tics with associated obsessive-compulsive disorders

The purpose of that study was to find the structural changes most correlated with each group. The results are summed up in Table 2.1:

	Cortical Thickness
Simple tics	Mostly found in primary motor regions
Simple and complex tics	Motor, premotor, prefrontal and parietal regions
Tics with OCD	Reduced cortical thickness in the anterior cingulate cortex

Table 2.1: Results of the work of Worbe et al. [WGH+10]

These results support "*the hypothesis that different symptom dimensions in Gilles de la Tourette syndrome are associated with dysfunction of distinct cortical areas*". The purpose of this project is to verify this hypothesis with the same data-set used in [WGH+10] but with a different methodology. Moreover, it will be also checked if there are some changes in the white matter tracts that link the cerebral cortex to the basal ganglia and constitute the main cortico-striato-thalamo-cortical circuits explained before.

Chapter 3

Data Set and first steps towards the Atlas construction

The aim of this chapter is to describe the data set used in this project and all the pre-processing steps applied to the raw data. The data set was kindly offered by the Institute NeuroSpin and in particular by Linda Marrakchi-Kacem and Cyril Poupon [MK11].

The first part will focus on a brief description of the methods used to acquire the data, MRI and DWI, and the first processing steps performed on them (by Linda Marrakchi-Kacem) to remove artifacts and extract the main structures such as cortical surface, basal ganglia nuclei and fiber bundles.

In the second half of this chapter, a definition of atlas construction will be given and all the pre processing steps related to it will be presented (conducted by the student). Moreover a general technique for bundle template initialization will be shown where it is kept the topology of fiber bundles which means the connectivity between the points of the fibers. As said before, one of the main goal of this project is to build an atlas for both surfaces and curves keeping also their topological structures. It can be seen as the sequel of Durrleman et al.'s paper [DPK+12] where they considered only surfaces and not curves. One of the main contribution of this project will be therefore to extend this atlas construction method also to fiber bundles.

3.1 Introduction and Description

The data set consists of T1 MRI (Magnetic Resonance Imaging) and DWI (Diffusion weighted Imaging) of 47 patients and 27 controls (16 males, 11 females, average age of 29.78±11.11 years). The patients were divided by the neurologists Dr Yulia Worbe and Dr Andreas Hartmann in 3 groups based on the complexity of the tics and on the presence or not of OCD (obsessive compulsive disorders) as described in the previous section [WGH+10]. The three groups were composed by:

- **Group 1**: 17 patients with simple tics . 11 males, 6 females, average age of 30.82±9.80 years.
- **Group 2**: 17 patients with simple and complex tics. 10 males, 7 females, average age of 33.00±13.63 years.
- **Group 3**: 13 patients with tics and OCD. 11 males, 2 females, average age of 29.46±8.72 years.

Severity of tics was assessed using the Yale Global Tic Severity Scale (YGTSS) [LRH+89] while the presence and severity of associated obsessive compulsive disorders was evaluated during the psychiatric consultation using the Yale-Brown Obsessive Compulsive Scale (Y-BOCS) assessment [GPRM89].

As described in [MK11], the results of the T1 MRI were initially processed with two software: FreeSurfer [fCNAAMCfBI12] and BrainVISA [dRnIN12] in order to extract the cortical surface and the basal ganglia. Instead, about DWI results, they were first treated to correct some artefacts using Connectomist 2.0 (a tool of BrainVISA) and afterwards a tractography method was used to extract fiber bundles of the whole brain [MK11].

3.1.1 T1-weighted MRI (Magnetic Resonance Imaging)

MRI is a radiologist technique used to visualize internal structures of the body in detail. It makes use of the property of nuclear magnetic resonance (NMR) to give as a result a 3D or 2D image, for example, of the brain. For this project the patients were treated with a strong static magnetic field of 3T using a Siemens Tim Trio 3T MRI system.

Every brain contains a lot of water and each water molecule has two hydrogen nuclei (protons) which have a spin equal to $+\frac{1}{2}$. Under the static magnetic field, the spin of the protons inside the brain start to align with the direction of the magnetic field in a parallel or anti-parallel way. In reality this alignment is never total but rather the spins start to show a precession around the direction of the magnetic field. This movement show a typical frequency, called Larmor frequency, which is in the field of Radio Frequency (3 kHz

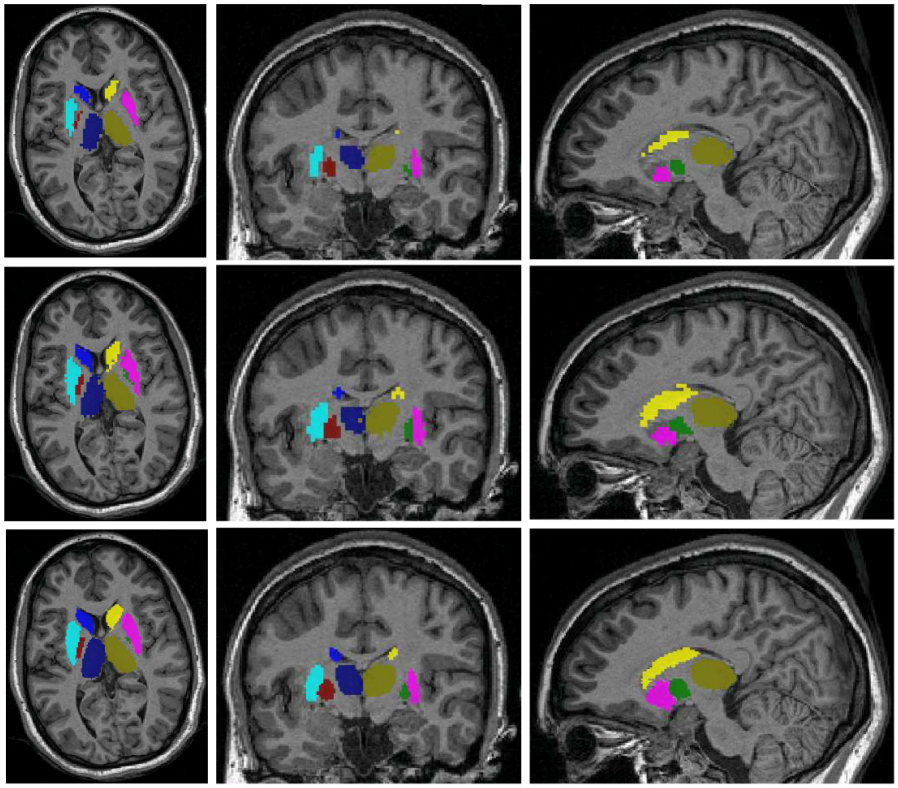


Figure 3.1: Example of a T1-weighted image with the segmentation of the internal structures (left and right caudate, putamen, globus pallidus and thalamus). Image taken from [MK11].

- 300 GHz) and in this case in the scale of MHz. If it is then produced an electromagnetic field at the same Larmor frequency, the protons will absorb the energy produced and their rotation will be enhanced by a specific angle, called flip angle. When the electromagnetic field is turned off, the spins of the protons will start to come back to their initial rotation around the direction of the magnetic field (relaxation). During this relaxation period a radio frequency signal is generated and it is measured on the plane perpendicular to the main magnetic field (Free Induction Decay or FID). At this moment it can be registered how the z component of the magnetization vector comes into thermodynamic equilibrium with its surroundings ("Spin-lattice relaxation"), which is characterized by a time constant called $\mathbf{T_1}$. Or how the transverse component of the magnetization vector decays towards its equilibrium value of zero ("Spin-spin relaxation") and in this case it is characterized by a time constant called $\mathbf{T_2}$. In reality, due to local inhomogeneities of the magnetic field, not all the proton spins gain the same velocity after switching on the electromagnetic field. Therefore it is usually used a 180 degree pulse to bring all the spins towards the main moment. This process will end up with a complete refocusing of the proton spins but, above all, with a deletion of the the effects due to inhomogeneities of the magnetic field. Moreover, when the horizontal component of all the momenta catch up with the

relative component of the main momenta (complete refocusing of the spins) it can be registered an echo from which the constant $\mathbf{T_2}$ can be measured. After the 180 degree pulse also the vertical components of the momenta reach the main momenta and the $\mathbf{T_1}$ constant can be computed since it characterizes the rate of this process.

Different tissues have different time constants and since $\mathbf{T_1}$ is significantly different between gray matter and white matter it is usually used for brain scans. Information about the origin of the signal in 3D space can be learned by applying additional magnetic fields during the scan. In fact, in this way, the strength of the magnetic field will vary depending on the position, making also the frequency of the released radio signal dependent on its origin in a predictable manner.

Sequence parameters were as follows: T1-weighted 3D MPRAGE, FOV=256mm, matrix 256x256, TE/TR=2.98ms/2.3s, TH=1.1mm, Phase FOV=93.8%, 160 slices per slab, RBW=240Hz/pixel. For more information about the parameters used during the scans please refer to [MKDG+12].

3.1.2 DWI (Diffusion weighted Imaging)

Diffusion-weighted magnetic resonance (MR) imaging provides image contrast that is different from the one given by conventional MR techniques. The aim of DWI is to provide images whose contrast is influenced by the differences in water molecules mobility.

Diffusion is a passive process driven by the ambient temperature and it is also known as Brownian motion. Since MRI is sensitive to the protons in water molecules, the main interest of neuroscience is in the diffusion of water molecules present in the brain and driven by body heat. In biological tissues, diffusion is not truly random because there are some structures as cell membranes or nerve fibers that constrain the movement of these molecules. Therefore it's not observed a "free" diffusion coefficient of water but an "apparent diffusion coefficient" (ADC).

In DWI, in addition to a strong static magnetic field present also in conventional MR techniques, two extra gradient pulses are also used which are equal in magnitude but opposite in direction and they are applied in a symmetric way with respect to the 180^o refocusing pulse. These two identical and opposite gradient pulses change linearly in the space. Therefore, if there isn't any movement in a tissue voxel, these two pulses cancel out since in each point there are two signals of the same magnitude but opposite direction. Instead, if the water molecules move in the same direction of the two gradient pulses (i.e. direction x), they are subjected to the first gradient pulse at one location and to the second one at a different position. The difference between the two gradient pulses is no more zero and it is proportional to the movement along the x-direction of the molecules.

The resulting signal differs from the one obtained using normal MRI by an amount related to the rate of diffusion which can be computed as follows:

$$\frac{S_{\mathrm{DW}}}{S_{\mathrm{MRI}}} = \exp(-b \text{ x } ADC) \tag{3.1}$$

where S_{DW} is the signal intensity on the DW image, S_{MRI} on the MRI image and b is the diffusion sensitivity factor which is proportional to the gradient amplitude, duration of the applied gradients and the time interval between paired gradients. Due to the high anisotropy of the brain the apparent diffusion coefficient is considered as a tensor. Since we are working in a 3D space, the three diagonal elements of the tensor give information about the magnitude of ADC in each dimension, while the off-diagonal elements show the correlations between the ADC values of different directions. In order to be able to study the anisotropic nature of diffusion in the brain (i.e. compute the tensor values) one needs to compare at least three different images obtained applying DW gradients in three orthogonal directions [SGG00].

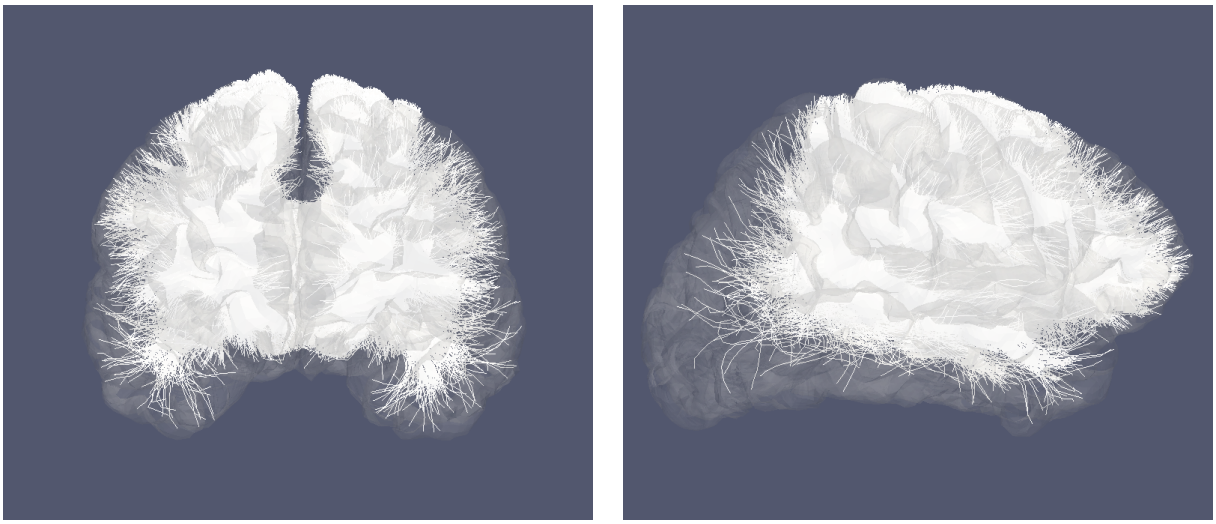
Sequence parameters were as follows: Single-shot twice refocused spin-echo DW-EPI, FOV=256mm, TH=2mm, matrix 128x128, TE/TR=86ms/12s, GRAPPA 2, partial Fourier 6/8, 80 slices, RBW=1630 Hz/pixel, b-value=1000s/mm^2, 50 directions. For more information about the DW Imaging please refer to [ST65], while more details about the parameters used during the scans can be found in [MKDG+12].

3.1.3 Data Processing

As described in [MKDG+12], cortical surface, basal ganglia nuclei (left and right caudate, putamen and globus pallidus) and left/right talamus can be extracted from the T1-weighted MRI using the automatic segmentation tools of FreeSurfer and BrainVISA. Information about the whole brain tracts (bundles) are taken both from the T1-weighted MRI and DW images. An Orientation Distribution Function (ODF)[1] field is computed for each subject using the analytical Q-ball model of Descoteaux et al. [DAFD07]. After that a whole brain streamline probabilistic tractography [PPC+05] is performed using this ODF field within a robust tractography mask obtained from the T1-weighted MRI data and registered to the DW data [MKDG+12]. The resulting "plate of spaghetti" is shown in Fig. 3.2. From the obtained whole brain tracts, the fiber bundles (set of tracts) linking each basal ganglia nuclei and thalamus to the cortex are selected obtaining 8 different bundles (4 left and 4 right). In order to do that, only the fibers having at least three points inside the considered nucleus and at least one point with a distance lesser than

[1] Approximation of the probability density function (PDF) of the average spin displacement of water molecules.

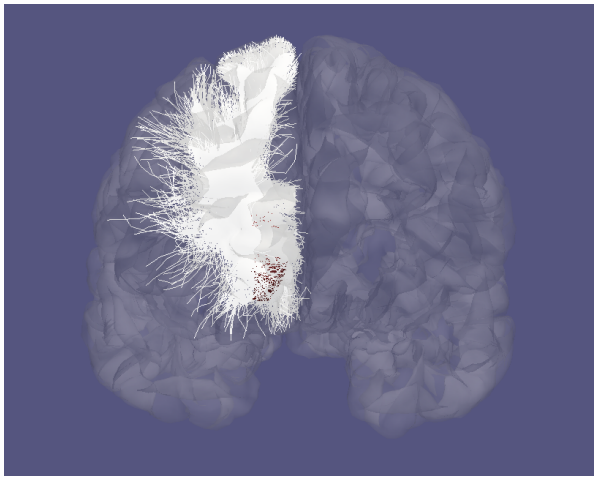
3mm from the cortical surface are considered. This process produces bundles which don't have any connexion with the contralateral hemisphere as it can be seen from Figures 3.3. In this project only 8 bundles are considered which connect Left and Right Caudate, Putamen, Globus pallidus and Thalamus to the Cortical surface. Moreover the fibers linking two different nuclei are not taken into account. Thus only a part of the cortico-basal ganglia-thalamus-cotical circuits is studied. The other links will be integrated in a future development of this study.



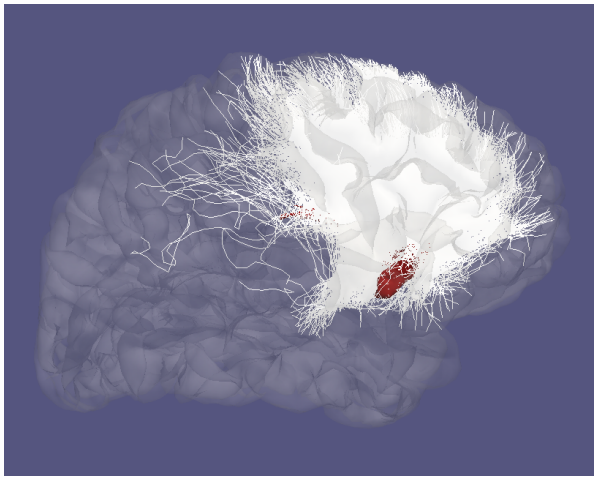
(a) Anterior view **(b)** Lateral view

Figure 3.2: Example of whole brain bundles.

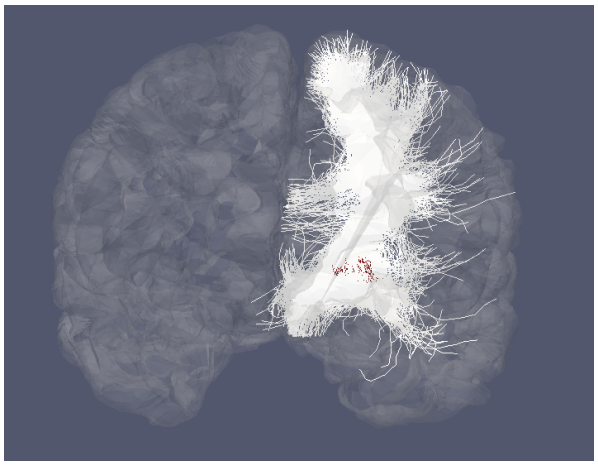
Furthermore, in order to be able to analyse surfaces from different subjects, a common spherical coordinate system is created [SRA+04]. This is based on the hypothesis that each hemisphere is topologically equivalent to a sphere as explained in [FSTD99]. Every cortical surface from all the subjects is inflated to a sphere and then warped to fit a spherical surface template so that the sulcal patterns (a specific part of the brain) match those of the template. The aim is to create a standard-mesh surface identical in geometry to the original one but with a topology shared among all the subjects. In order to do that it's first build a triangulated icosahedron (with 81924 nodes in this project) whose co-centred sphere has a radius equal to the one of the template sphere. Each node of the icosahedron, projected radially onto the warped sphere, falls into a triangle. It's important to notice that the warped sphere is exactly the original mesh inflated to a sphere and after modified so that it has the same radius as the template sphere. Therefore it's still divided in triangles as a mesh file.
After that, it's assigned to each icosahedron node a coordinate value for each dimension, as explained in Fig.3.4 this is done using the following set of equations:



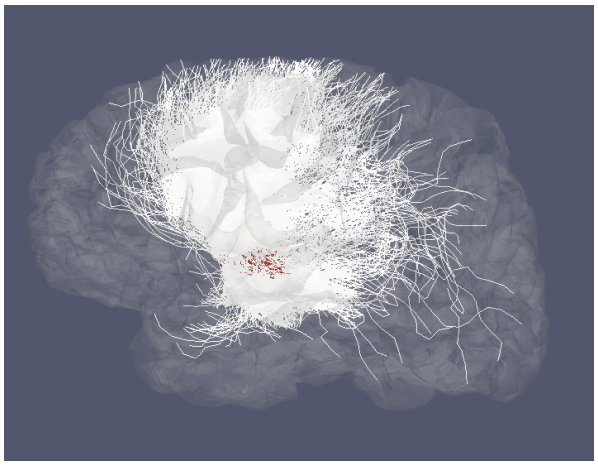
(a) Left Caudate and respective bundle, anterior view



(b) Left Caudate and respective bundle, lateral view



(c) Right Globus Pallidus and respective bundle, anterior view



(d) Right Globus Pallidus and respective bundle, lateral view

Figure 3.3: Examples of internal structures and relative bundles.

$$x_n = a_1 x_{n1} + a_2 x_{n2} + a_3 x_{n3} \tag{3.2}$$

$$y_n = a_1 y_{n1} + a_2 y_{n2} + a_3 y_{n3} \tag{3.3}$$

$$z_n = a_1 z_{n1} + a_2 z_{n2} + a_3 z_{n3} \tag{3.4}$$

Each coordinate of the icosahedron node is a weighted sum of the coordinates of the vertices of the triangle of the warped sphere where the icosahedron node falls into. The weight relative to a vertex is the area of the triangle given by the barycentre and the opposite two vertices. Eventually one obtains a cortical surface with the same number

of vertices and triangles of the icosahedron but with a structure (geometry) similar to the initial mesh. Therefore this can be seen as an interpolation scheme to compute the coordinates of the standardize cortical surface using the original ones.

Using this procedure we obtained a set of 74 cortical surface meshes sharing the same number of vertices and triangles. This permits to greatly simplify the inter-subject surface-based analysis as explained in [SRA+04]. However, in this project, since we are using the framework of currents, we don't need to have necessarily the same number of vertices and faces between two different cortical surfaces. In fact, as explained in Chapter 1, the framework of currents is robust to different sampling.

Anyway, all the processing steps shown so far throughout Chapter 3 were carried out by Linda Marrakchi-Kacem and not by the student. We described them for the sake of completeness since the data received at the beginning of this project were exactly the results of all these procedures.

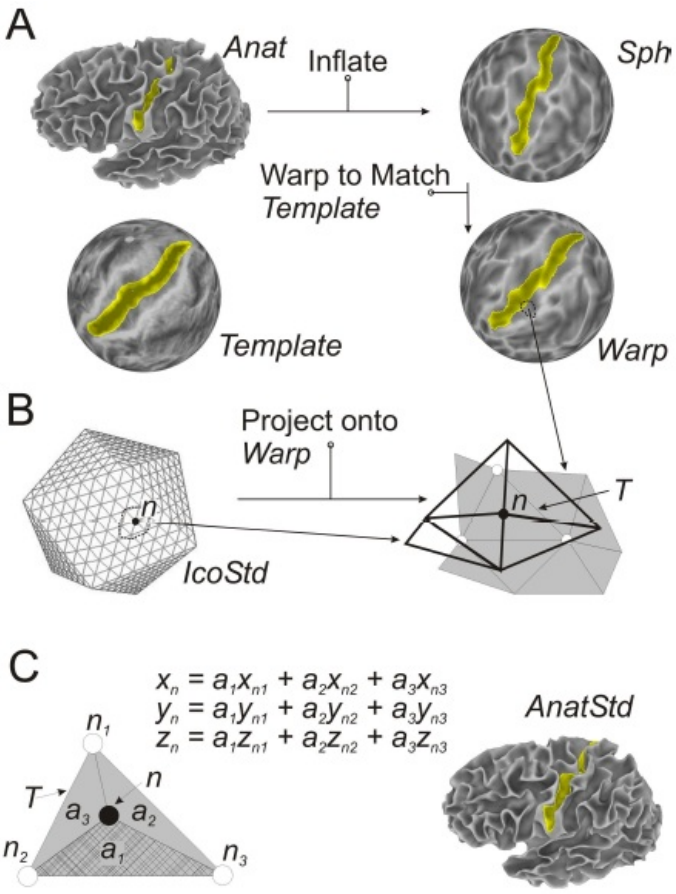


Figure 3.4: Inflating and warping of cortical surface (*Anat*) to spherical template and creation of a standard-mesh (*AnatStd*) with the sulcal patterns highlighted in yellow. Image taken from [SRA+04].

3.2 Towards the atlas construction

To sum up what explained so far, for each one of the 74 subjects present in the data set, there are:

1. 1 cortical surface saved as 3D mesh (".mesh" in Anatomist/BrainVisa)[2].
2. 8 shapes saved as 3D mesh of left and right Caudate, Putamen, Globus Pallidus and Thalamus.
3. 8 fiber bundles saved as 3D segment sets (".bundlesdata" in Anatomist/BrainVisa) representing the tracts linking left and right Caudate, Putamen, Globus Pallidus and Thalamus to the Cortical Surface.

All the ".mesh" files are polygon mesh which are constituted by a collection of vertices and triangular faces and they approximate the shape of the 9 objects listed before. The set of vertices is the collection of all the points while the faces are defined by 3 index representing the 3 vertices of each triangle. The ".bundlesdata" files are collections of 3D curves. Each file is constituted by a set of points and, for each curve, a series of index. Each index is linked to one point.

All the mesh files come from the T1 frame while the fiber bundles are extracted from the DW frame. Using a tool of BrainVisa/Anatomist it is possible to compute the Rotation matrix and the Translation vector of the transformation to go from one referential to the other. In this way all the bundles points are converted in the T1 frame.

3.2.1 Definition of Atlas construction

The construction of an atlas from a set of anatomical shapes of different subjects belonging to the same population is a process based on two main steps [DPTA08]:

1. Construct a template (or "mean anatomy") for each anatomical object inside the population.
2. Estimate the variations of these templates within the population by deforming them into each subject's object present in the population.

[2]Where Anatomist is just a visualisation software of the files produced by BrainVisa.

Usually, in order to avoid biases introduced by separate processing, these two operations are carried out together and so the template as well as its deformations are jointly estimated.

In this project, using as source of inspiration [DPTA08] [MMTY08] and [AAT07], the statistical estimation of the atlas is based on a *forward model* which considers the observations (T_i) as noisy deformations $(\phi_i * T_i)$ of an unknown template $(\overline{T})$, and it can be written as:

$$T_i = \phi_i * \overline{T} + \epsilon_i \quad \Leftrightarrow \quad \overline{T} = \phi_i^{-1} * T_i - \phi_i^{-1} * \epsilon_i \tag{3.5}$$

where ϵ_i are i.i.d. random variables (i.e. Gaussian), whereas the *backward model* is defined as:

$$\phi_i * T_i = \overline{T} + \epsilon_i \quad \Leftrightarrow \quad T_i = \phi_i^{-1} * \overline{T} + \phi_i^{-1} * \epsilon_i \tag{3.6}$$

which means that, in contrast with the forward model, the template is an average of deformed observations (see Fig.3.5). Assuming that it is possible to define probabilities on the shapes and deformations, one is interested in both finding the probability of having a template given a training database $\{T_i\}$: $p(\overline{T}|\{T_i\})$, and computing the likelihood of a new observation T_{new} given the template $\overline{T}$: $p(T_{new}|\overline{T})$. Looking at Eq.3.5 and 3.6 it is trivial to notice that these two passages have a reverse difficulty in the two methods. In fact, if one uses a Gaussian distribution for the noise ϵ_i, the distribution of $p(\overline{T}|\{T_i\})$ is also Gaussian using the backward model while it follows the distribution of $\phi^{-1}\epsilon_i$ in the forward model becoming therefore more difficult to estimate. About the likelihood of a new observation $p(T_{new}|\overline{T})$, the result is exactly the opposite with an easier computation in the forward model. Since it is better to spend more time to build the atlas, which is done once for all at the beginning of the process, and to consume less time for each new observation, the forward model seems to be more suitable from a computational point of view.

When dealing with a *forward model* one would like also to find the "best" template. In this process there are two main passages: at the beginning all the observations T_i are jointly used to mould the template and after the new template $\overline{T}_0$ is used to compute all the deformations ϕ_i to the T_i. In this way it's possible to find a template which captures the common features across the population. More information about the atlas construction and about the kind of deformation used in this project will be given in Chapter 4.

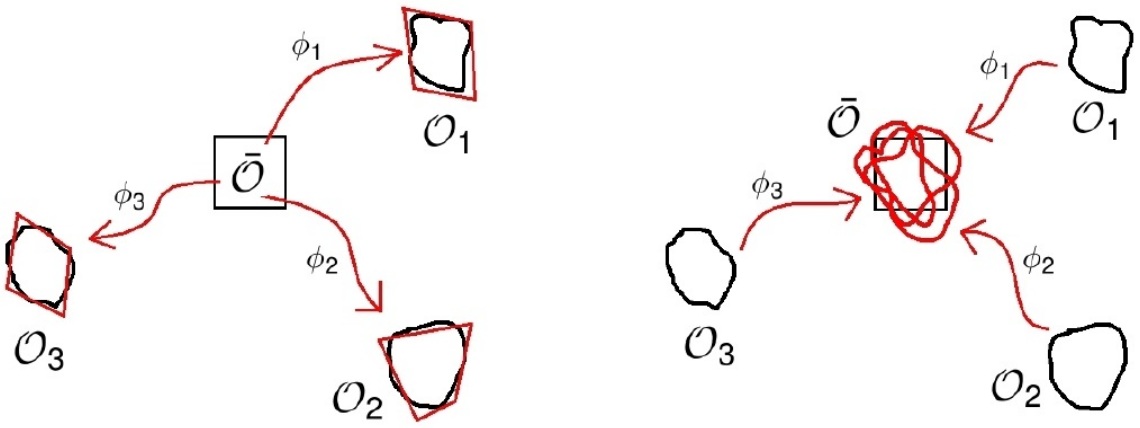
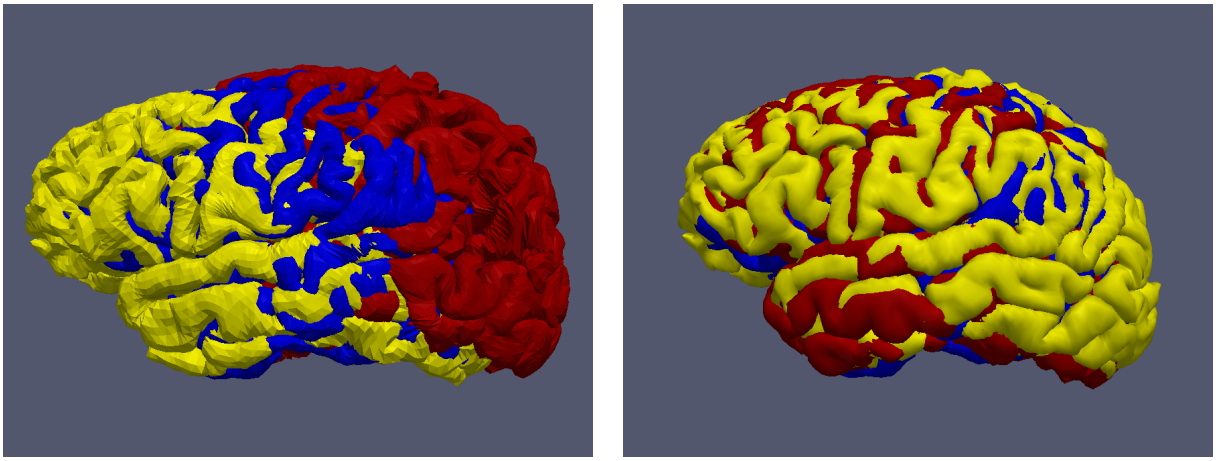


Figure 3.5: Forward scheme model on the left and Backward scheme model on the right taken from [DPTA09].

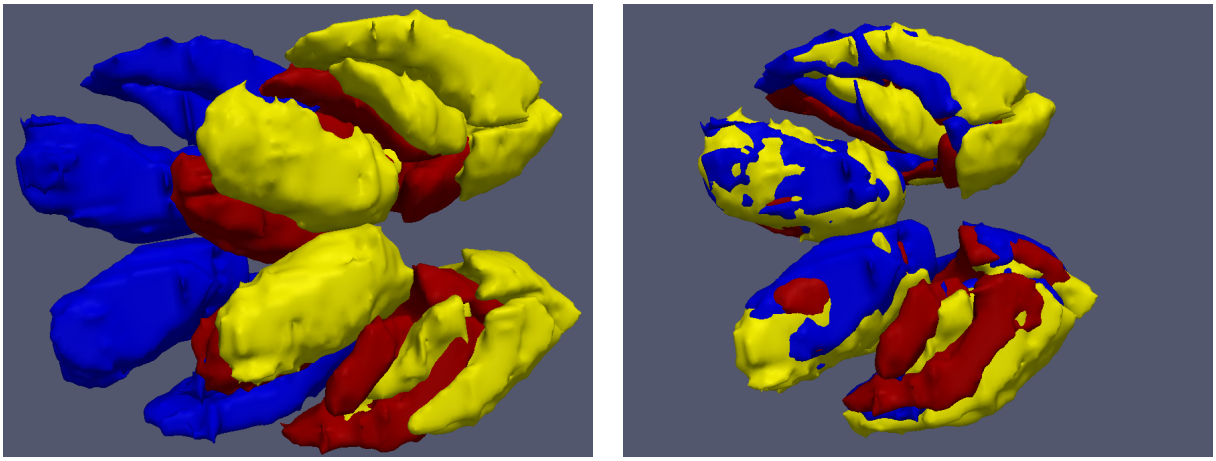
3.2.2 Pre processing for the Atlas construction

Centring During the atlas construction, for each kind of object, the template is deformed and matched to each object within the population. This is the technique used in this project to study the variability of the template which can be seen as a characteristic of the population. The position of the template is initialized as an average of the positions of the objects. Therefore, if the objects are far from each other, the matching procedure, which can be seen as a minimization process, requires a lot of energy (and time) just to superimpose the template onto the object. A possible solution to this problem is to reduce the distances between the objects. The first step is to have all the elements of the brains of each subject not only in the same frame, but also around the same midpoint.
Thus, every brain is centred subtracting the mean along every dimension from all the points. The mean is computed considering only the mesh files (internal objects and cortical surface) while it is subtracted from all the objects, both mesh and bundles. The reason of this choice is that we want to center around 0 mainly the cortical surfaces in order to eliminate the "artefact" created by the different positions adopted by the subjects during the scans. The fiber bundles are not used in the computation of the mean because their estimation was based on a probabilistic method and so they might have more points in a part of the brain with respect to the others. This can influence the centring process making it less reliable since the number of points of the bundles is similar to the one of the cortical surface.
This centring process is therefore carried out to decrease the distances between the same kind of objects of different subjects keeping the relative distances between the objects and the cortical surface.



(a) Before centring **(b)** After centring

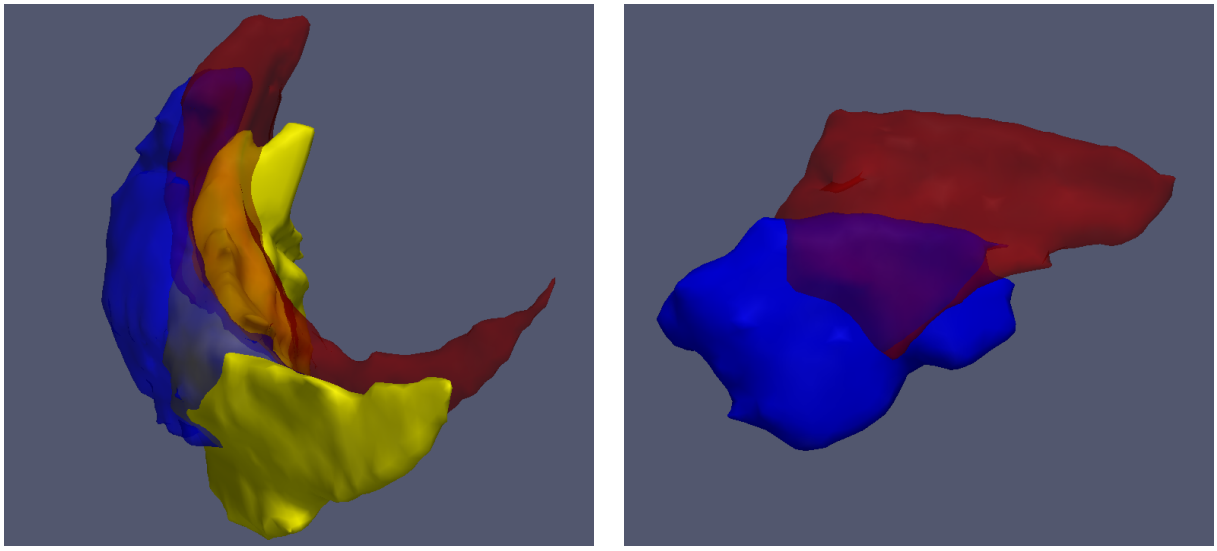
Figure 3.6: Example of cortical surfaces before and after being centred.



(a) Before centring **(b)** After centring

Figure 3.7: Example of internal structures before and after being centred.

Rigid Registration After that, as it is possible to see from Fig.3.6, the cortical surfaces of the subjects are all aligned while the internal structures are still a bit far from each other (see Fig.3.7 and Fig.3.8). This is because the number of points of the cortical surface is about 4 times bigger than the number of points of the internal objects (81924 Vs 20500 on average). Therefore the centring process regards more the cortical surface than the basal ganglia nuclei.
A send step is then needed to push also the internal objects close together. We decided to use a rigid registration between all the objects and a target one chosen as reference (subject 1, in this project). All the objects of every subject are moved in a rigid way, which means rotation and translation, in order to be more "aligned" and "better superimposed"



(a) Left Caudate after centring **(b)** Left Globus Pallidus after centring

Figure 3.8: Example of internal structures after being centred.

with the objects of the first subject. This process reduces the distances between the same kind of objects and therefore also the distances between the template and each object. The choice of aligning every object with respect to the ones of the first subject was purely random.

For this rigid registration we decided to use a model based on Gaussian mixtures whose Matlab and C++ implementation was freely available at *http://gmmreg.googlecode.com* [JV11]. In most of the rigid registration schemes one wants to find the transformation that best aligns the "source" shape to the "target" shape. Usually the two objects can be seen as sets of unstructured points which don't have any spatial information or label. In the paper of Jian et al. [JV11], the authors investigate the idea of representing discrete point sets by continuous density functions, namely Gaussian mixture models (GMM). The probability density function of a general Gaussian mixture is defined as:

$$p(\mathbf{x}) = \sum_{i=1}^{N} w_i \phi(\mathbf{x}|\mu_i, \Sigma_i) \tag{3.7}$$

where

$$\phi(\mathbf{x}|\mu_i, \Sigma_i) = \frac{\exp\left[-\frac{1}{2}(\mathbf{x}-\mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i)\right]}{\sqrt{(2\pi)^d |\det(\Sigma_i)|}} \tag{3.8}$$

and $\mathbf{x}$ is a vector of points in $\mathbf{R}^d$ space while the other elements are:

- N, the number of Gaussian components.
- w_i, the weights.
- μ_i, the mean vectors.
- Σ_i, the covariance matrix.

As suggested in [JV11], we decided to approximate both the "source" shape and the "target" shape using one component for each point present in the set, weighting all the components in the same way (w_i=w), using also the same spherical covariance matrix Σ_i=Σ for all the components (i.e. a diagonal matrix proportional to the identity matrix) and settings the mean vectors μ_i equal to the spatial coordinates of each point. The resulting density is an over-parametrized Gaussian mixture which can be modified clustering or grouping some points which come from the same Gaussian distribution. In this project, since the number of points wasn't excessive and therefore also the computational time, the clustering method wasn't applied.

After having approximated both the "source" shape and the "target" shape as GMM an optimization procedure is carried out using as dissimilarity measure a L2 distance since it is strongly related to the inherently robust estimator L_2E [JV11] and also because it exists a closed-form expression for the L2 distance between Gaussian mixtures. Formally, given two point sets: S for the source shape and T for the target shape, this registration method finds the parameters θ of the rigid transformation V which minimizes the following cost function:

$$d_{L_2}(S, T, \theta) = \int (\mathrm{GMM}(S) - \mathrm{GMM}(V(T, \theta)))^2 dx \tag{3.9}$$

where writing $f(\mathbf{x}) = \mathrm{GMM}(V(T, \theta))$ and $g(\mathbf{x}) = \mathrm{GMM}(S)$ it can also be written as:

$$\int f^2(\mathbf{x})dx + \int g^2(\mathbf{x})dx - 2\int f(\mathbf{x})g(\mathbf{x})dx \tag{3.10}$$

and since $g(\mathbf{x})$ is fixed during the optimization and $\int f^2(\mathbf{x})dx$ is invariant under rigid transformations, the only part updated during the procedure is the inner product between the two GMMs. The transformed set of target points T and the original set T_0 are related by the rigid motion: $T = \mathbf{R}T_0 + \mathbf{t}$ where $\mathbf{R}$ is the rotation matrix and $\mathbf{t}$ is the translation vector. Therefore the parameters θ of the transformation V are exactly the elements of $\mathbf{R}$ and $\mathbf{t}$. Moreover, noting that the integral of the product of two Gaussian component densities is : $\int \phi(\mathbf{x}|\mu_1, \Sigma_1)\phi(\mathbf{x}|\mu_2, \Sigma_2)dx = \phi(0|\mu_1 - \mu_2, \Sigma_1 + \Sigma_2)$, one can write $\int f(\mathbf{x})g(\mathbf{x})dx$ in a closed form. In fact, if $f(\mathbf{x}) = \sum_{i=1}^{M} \alpha_i \phi(\mathbf{x}|\mu_i, \Sigma_i)$ and $g(\mathbf{x}) = \sum_{j=1}^{N} \beta_j \phi(\mathbf{x}|\gamma_j, \chi_j)$

are two points sets (target and source respectively) modeled by Gaussian mixtures, the transformed target set can be written as: $f_V(\mathbf{x}) = \sum_{i=1}^{M} \alpha_i \phi(\mathbf{x}|\mathbf{R}\mu_i + \mathbf{t}, \mathbf{R}(\Sigma_i)\mathbf{R}^T)$ and their inner product is:

$$\int f(x)g(x)dx = \sum_{i=1}^{M}\sum_{j=1}^{N} \alpha_i \beta_j \phi(0|\mathbf{R}\mu_i + \mathbf{t} - \gamma_j, \mathbf{R}(\Sigma_i)\mathbf{R}^T + \chi_j) \tag{3.11}$$

This last equation is the cost function of the optimization procedure and it is also differentiable with respect to the rotation and translation parameters. Jian et al. [JV11] found out that both a gradient descend scheme and a non-stochastic gradient-free optimization method, such as the Powell's method, are usually sufficiently fast to achieve quite accurate results. But in the Matlab implementation of the algorithm they suggested to use the function *fminunc* with a large scale optimization. This algorithm is a subspace trust-region method and it is based on the interior-reflective Newton method. For more information about this algorithm please check the Matlab help. About the stopping criteria we used 100 as maximum number of function evaluations allowed and 10^{-10} for the termination tolerance on both the function value and on the parameters value. The whole algorithm can be summed up in the following way:

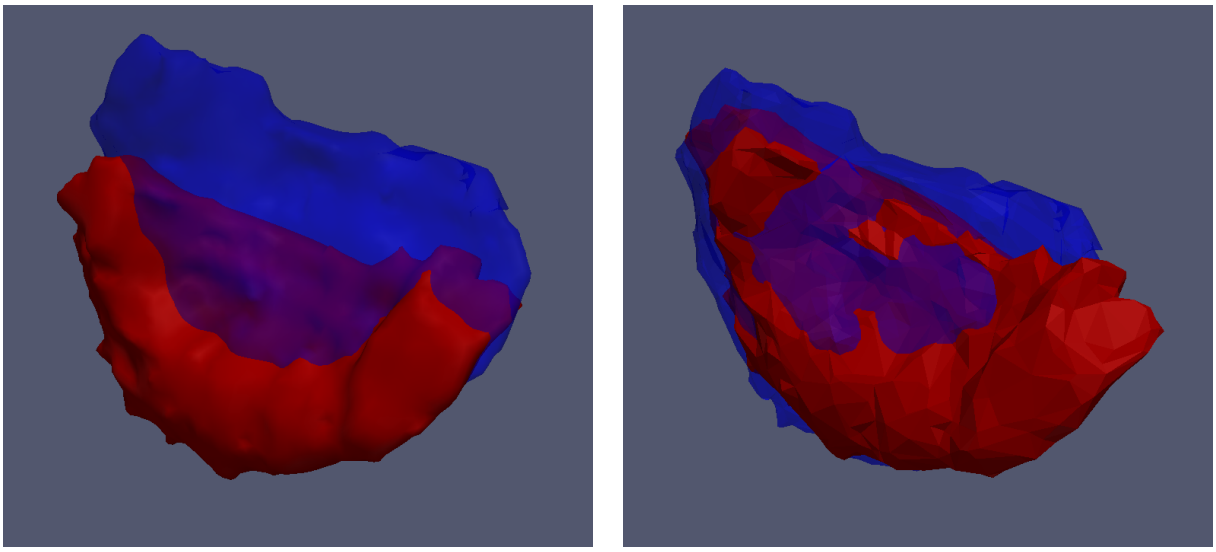
1. Approximate both source and target shape with a GMM.
2. Specify initial parameters θ (i.e. identity transform).
3. Set up the L2 distance between the two GMMs.
4. Optimize the objective function with a gradient descend scheme (or a Powell's scheme).
5. Repeat the last point until a certain stopping criteria is satisfied.

This method works also for non-rigid transformations but in this project it was used only for the rigid one. For more information about this algorithm please refer to [JV11] and to [JV05].

As it can be seen from Fig.3.9 and Fig.3.10 this Rigid Registration scheme achieved the expected result. In fact all the objects are now better aligned with the objects of the first subject and therefore they have all a similar position but they haven't been deformed since the transformation was rigid.

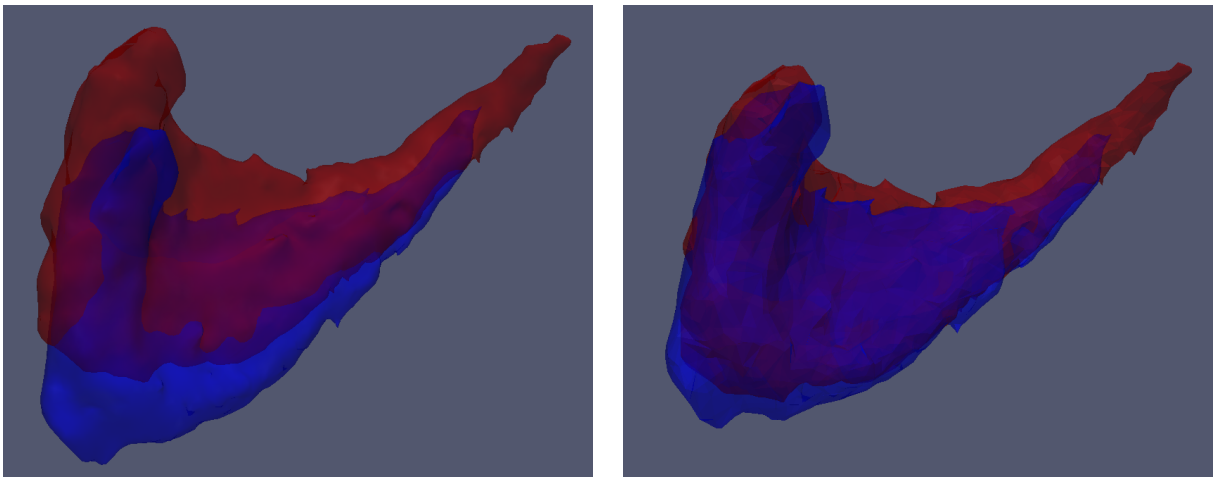
As it is shown in the next chapter, we are looking for a specific non-rigid deformation that best matches a common template to all the objects of different subjects. The rigid registration is useful to reduce the amount of possible configurations that the deformation

procedure must take into account. Translations and rotations are related only with the relative positions between brains of different subjects. But we are more interested in the non-linear features of the shapes of the objects and not in their positions.



(a) Left Putamen after centring **(b)** Left Putamen after registration

Figure 3.9: Example of left putamen after rigid registration.



(a) Left Caudate after centring **(b)** Left Caudate after registration

Figure 3.10: Example of left caudate after rigid registration.

Bundles Template Initialization The atlas construction procedure uses a gradient descend scheme to estimate both the templates of each kind of object and their deformations into each object of the population. Therefore it is very important the initialization

of the template for each kind of object. As written before, the goal of this project is to extend the atlas construction method used in [DPK+12] (only with surfaces) also for curves (fiber bundles). The principal characteristic is that the topology of the objects must be preserved also in the templates. Thus the bundle template must keep the connectivity between the points of the fibers.

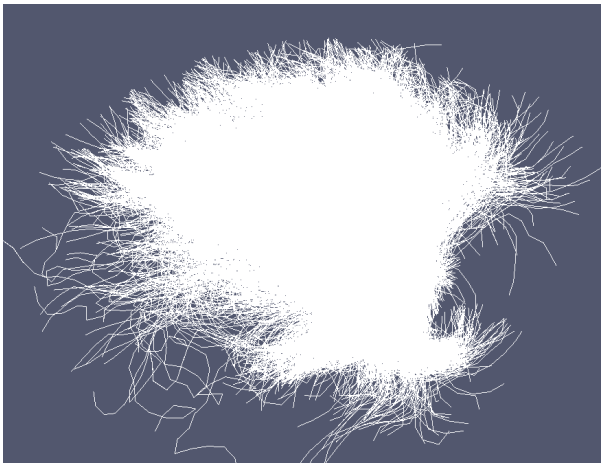
In this project the number of types of bundles is 8. Therefore one needs to have 8 bundles templates which start from 6 different basal ganglia nuclei (left and right caudate, putamen and globus pallidus) and also from the left and right thalamus, ending in one of the two hemisphere of the cortical surface.

The initialization of a template for the bundles is pretty complicated since they are constituted by thousands of fibers which show a great variability among the population. Therefore one is tempted to investigate the global shape of the whole bundle which seems more stable (less variable) between different subjects. But, in order to keep the same topology, one is obliged to build the template with fibers. A simple idea could be to select for each kind of bundle an area in the cortical surface and in the correspondent starting nucleus and to initialize the bundle with straight lines (sets of segments) connecting these two areas. The problem linked to this method is that, while the part of the nucleus from which the fibers start is not so spread, the receiving area in the cortical surface is really big and it's difficult to find a rule with which decide from where the fibers should start and where they should end. Moreover the number of fibers for each bundle is really high (3000-20000 fibers per bundle).
Thus we decide to employ another method. As first step we reduce the amount of data of each bundle by keeping randomly only 10% of the fibers using an uniform random distribution. In this way, as it is possible to see from Fig.3.11, there is not only a benefit from a computational point of view (less points to analyze) but also for a better comprehension since it seems to be more enhanced the internal structure of the bundles. Therefore it can be easier to check if the bundle template preserves the common shape of the population.

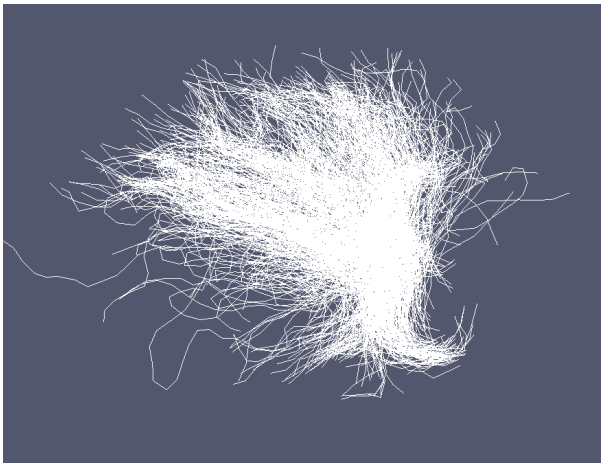
After reducing the number of fibers in each bundle of every subject we decide to create a bundle template in the following way:

For each kind of bundle
- Select randomly 2% of the fibers from the initial bundle of each subject (before the random selection) using a constrain.
- Discard a certain number of fibers of the bundle obtained after the previous step using a greedy approximation method based on the framework of currents.

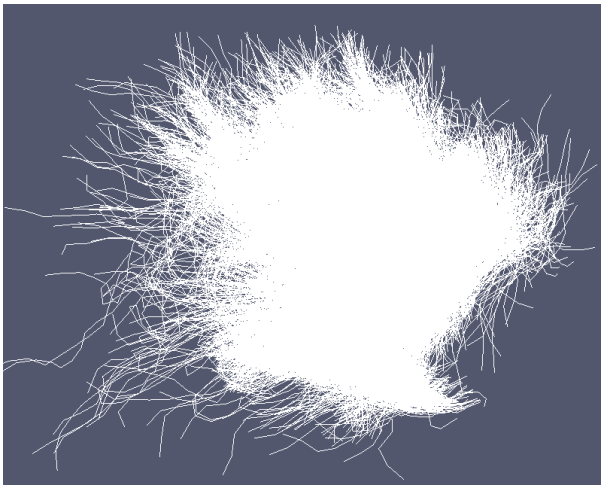
In this way we can create a template which preserves the same topology of the bundles and we can also regulate its number of fibers.
In the first step we use a uniform random distribution to select a fiber from a subject bundle but we keep this fiber in the template bundle only if the following condition is fulfilled:



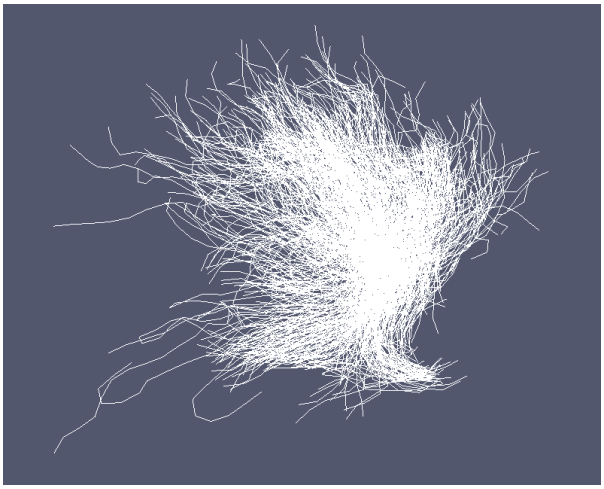
(a) Initial "left putamen" bundle



(b) 10% of fibers random selected from the "left putamen" bundle



(c) Initial "right putamen" bundle



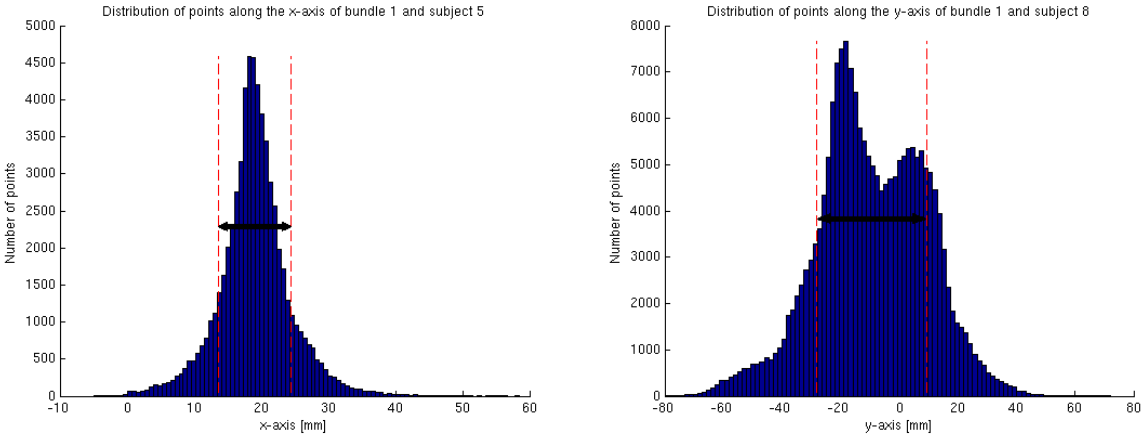
(d) 10% of fibers random selected from the "right putamen" bundle

Figure 3.11: 10% of fibers uniformly random selected from two different bundles.

Each selected fiber has to pass through a parallelepiped whose sides are equal to the mean ± std of the 3 projections of the subject's bundle onto the three Cartesian axes x,y,z

The aim of this procedure is to keep only the fibers which have at least one point inside the "most well-travelled part" of the bundle, trying therefore to reject all the fibers which are far from the main anatomical "waves" of the bundle considering them as outliers. Moreover this technique seems to be quite correct since, using a visual inspection, a lot of projections mimic an unimodal normal distribution as it is possible to see from Fig.3.12. These projections are exactly the distributions of each coordinate of the points of the

bundles. Therefore the utilization of the mean ± one standard deviation seems to be quite reasonable since it considers as outliers the points less present in the bundle and it covers 68.2% of the data. But at the same time some of the projections present a bimodal behaviour which can't be well approximated using the mean ± one standard deviation. Therefore it could be used also here a Gaussian mixture model to approximate the projections and another rule to select the fibers but, due to lack of time, we decide to leave it as a future development. Nevertheless this method seems to work quite fine since it rejects the fibers more aside of each bundle creating a big and dense initial template as it is possible to see from Fig.3.13.

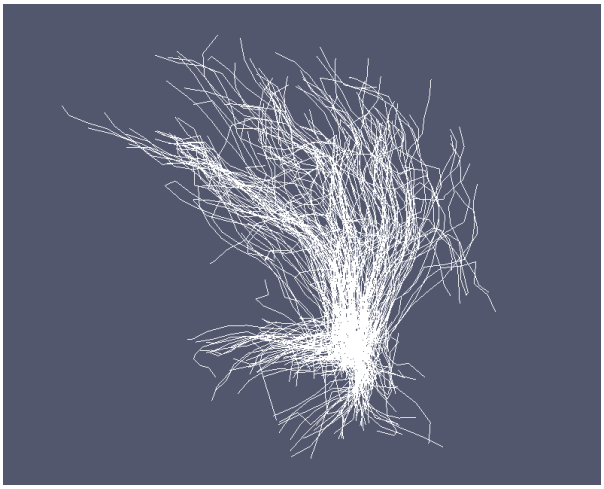


(a) Example of unimodal distribution **(b)** Example of bimodal distribution

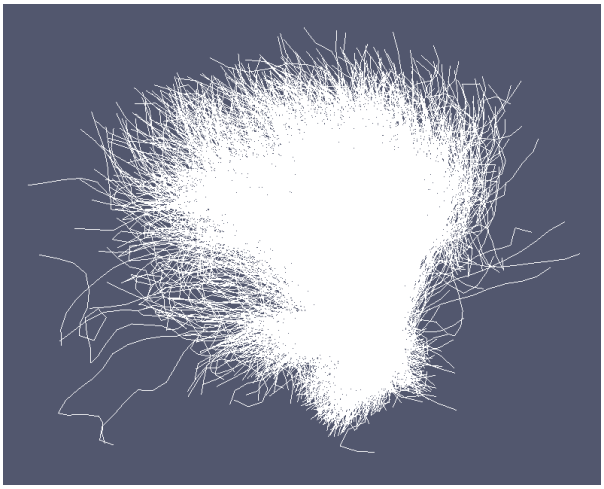
Figure 3.12: Distributions of points of the first bundle of subject 5 in the x-axis and of subject 8 in the y-axis respectively. The first one is unimodal while the second one seems to be bimodal. The black arrow represents the interval between the mean ± one standard deviation.

At the end of the first step we have 8 prototypes of bundle template, one for each kind. Since the number of subjects is 74 and we have selected 2% of the fibers from each subject, these 8 prototypes are characterized by a number of fibers equal to $\sum_{i=1}^{74} 0.02 * B_i$ where B_i is the number of fibers of a bundle of one subject before the 10% random selection. Of course we would like to have a bundle template with a number of fibers similar to the average one considering all the bundles after the 10% random selection. Therefore we want to reduce the number of fibers in each bundle template until it reaches $\frac{1}{74}\sum_{i=1}^{74} 0.1 * B_i$. Thus the second step regards the selection of the most important fibers of these prototypes until their number of fibers is equal to the average number of fibers considering all the bundle after the 10% random selection. In order to do that, it is used a greedy approximation method for each type of bundle:

- It is first computed, in the framework of currents, the norm of the difference between the bundle template and 50 fibers belonging to the same bundle template.



(a) 10% of fibers of the right globus pallidus bundle of subject 58



(b) 2% of fibers from each subject

Figure 3.13: Comparison between a bundle of a subject and the result of the first step of the bundle template initialization.

- This process is carried on for all the fibers of the bundle template selecting every time 50 different fibers.
- The norm of the difference is stored at each step.
- The 50 fibers linked with the minimum norm of the difference are saved in a new bundle template and removed from the initial bundle template.
- The process is repeated until the new bundle template reaches a certain number of fibers which is, in this project, the average number of fibers considering all the 10%-reduced bundles of every subjects.

From a mathematical point, one wants to minimize the following equation:

$$\min_{k} \|T - \sum_{i=(k*50)+1}^{(k+1)*50} F_i\|_{W*} \qquad k = 0 : \left(\frac{N}{50}\right) - 1 \tag{3.12}$$

where N is the total number of fibers of the initial bundle template which is updated ($N = N - 50$) every time that 50 fibers are selected and saved in the new bundle template. Eq.3.12 can also be rewritten in the following way:

$$\begin{aligned}
\min_k \|T - \sum_{i=(k*50)+1}^{(k+1)*50} F_i\|^2_{W*} &= \|T\|^2_{W*} + \|\sum_{i=(k*50)+1}^{(k+1)*50} F_i\|^2_{W*} - 2\langle T, \sum_{i=(k*50)+1}^{(k+1)*50} F_i\rangle_{W*} \\
&= \langle T, T\rangle_{W*} + \langle \sum_i F_i, \sum_i F_i\rangle_{W*} - 2\langle T, \sum_i F_i\rangle_{W*} \\
&= \langle L_W^{-1}(T), L_W^{-1}(T)\rangle_W + \langle L_W^{-1}(\sum_i F_i), L_W^{-1}(\sum_i F_i)\rangle_W \\
&- 2\langle L_W^{-1}(T), L_W^{-1}(\sum_i F_i)\rangle_W \\
&= L_W(L_W^{-1}(T))(L_W^{-1}(T)) + L_W(L_W^{-1}(\sum_i F_i))(L_W^{-1}(\sum_i F_i)) \\
&- 2L_W(L_W^{-1}(T))(L_W^{-1}(\sum_i F_i)) \\
&= T(L_W^{-1}(T)) + \sum_i F_i(L_W^{-1}(\sum_i F_i)) - 2T(L_W^{-1}(\sum_i F_i)) \\
&= \sum_{j=1}^{N} \alpha_j^T \gamma_T(x_j) + \sum_{i=1}^{M} \beta_i^T \gamma_F(y_i) - 2\sum_{j=1}^{N} \alpha_j^T \gamma_F(x_j) \qquad (3.13)
\end{aligned}$$

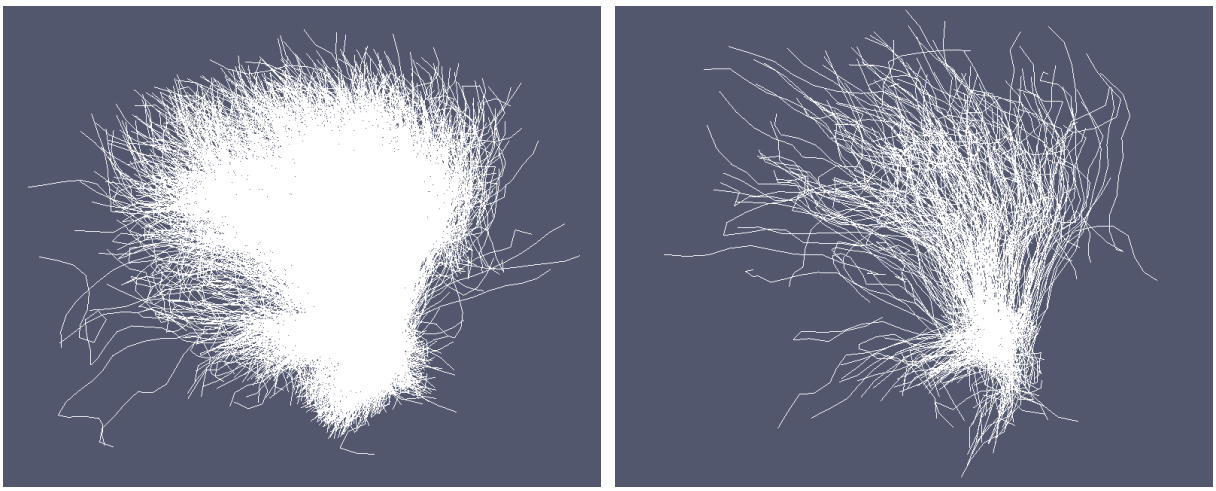
where $T = \sum_{j=1}^{N} \delta_{x_j}^{\alpha_j}$ and $\sum_i F_i = \sum_{i=1}^{M} \delta_{y_i}^{\beta_i}$ and $\gamma_T = L_W^{-1}(T)$ is the vector field related to the bundle T while $\gamma_F = L_W^{-1}(\sum_i F_i)$ is the vector field linked to the 50 fibers under examination. Throughout the passages it was used both the definition of currents in Eq.1.3 and the definition of mapping between the space of vector fields W and its dual space (the space of currents) $W*$ in Eq.1.10.

The computation of Eq.3.13 requires at each step the vector field of the bundle γ_T and the vector field of the 50 fibers under examination γ_F. γ_T is computed once for all at the beginning of the process while γ_F is computed at the beginning of each iteration. After that it's also necessary to use the interpolation scheme PVI (explained in section 1.3.1) to find the values of the vector field γ_T in the points of the 50 fibers. The longest (in time) computation is the estimation of the vector field of the bundle since its associated grid is usually very big and the number of points is relatively large (usually the dimension of the grid is [512,512,512] points). Moreover, being a greedy approximation, not all the possible combinations of 50 fibers are estimated but at each step 50 fibers are taken randomly and each fiber is used only in one iteration. Nevertheless the time for finding one bundle template is about 1-2 days, depending on the size of the bundle and on the server used. For example, with an initial bundle template of 13597 fibers, it took about 34 hours to complete this greedy approximation method finding the "final" bundle template of $\frac{13597*0.1}{0.02*74} = 918$ fibers, approximated to 900 since we analyse 50 fibers at each iteration. The specification about the server used are: Intel (R) Xeon(R) model 44, CPU X5650 @

2.67GHz with cache size 12288 KB.

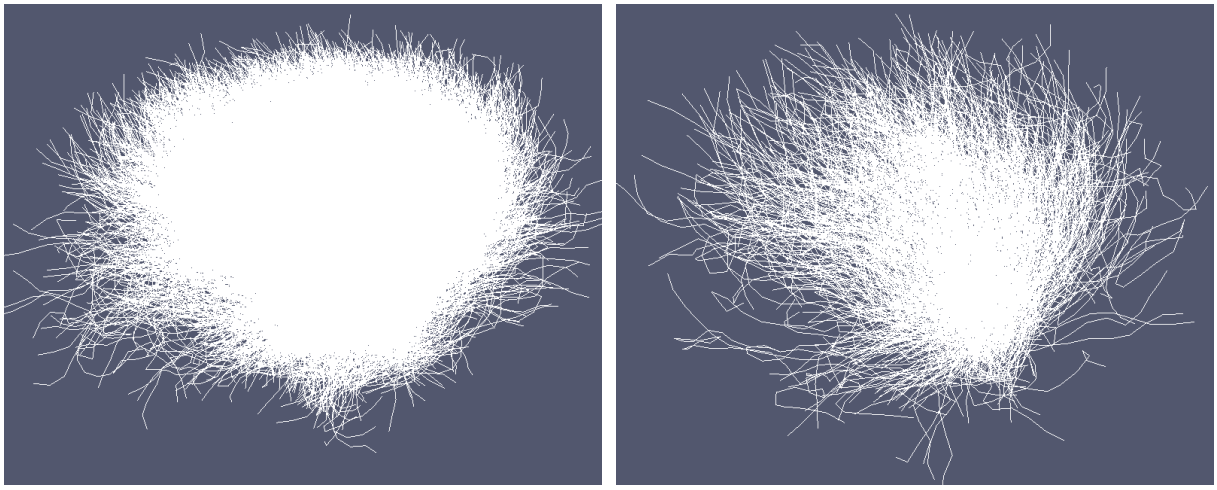
Furthermore it's important to notice that only the two last terms of Eq.3.13 differ at each iteration, and since we are interested in the differences between the results of each iteration, the first term $\|T\|^2_{W*}$ is not computed. Other techniques for bundle template estimation were tested but we obtained the best results, in terms of visual inspection and time consumed, with the one explained before.

Two examples of bundle template are shown in Fig.3.14 and in Fig.3.15 for the right globus pallidus bundle and right putamen bundle respectively. The amount of fibers in the final template is equal to the mean value of the number of fibers of the 10%-reduced bundles considering all the subjects. The resulting bundle is therefore similar to the ones obtained after the 10% reduction, as it is possible to notice comparing Fig.3.13a and Fig.3.14b. A scheme of the template initialization process for each kind of bundle is presented in Fig.3.16. The two columns at the two extremities represent the bundles of each subject. In the left part it is shown the result of the first step with the bundle composed by 2% of fibers from each subject bundle. Instead, in the right part, there are all the resulting bundles after selecting 10% of the fibers from each subject bundle. In the central part it is shown the resulting Initial template after the second step of our algorithm. It can be noticed that, also from a visual inspection, its number of fibers is similar to the ones of the subject bundles after the 10% selection.



(a) 2% of fibers from each subject **(b)** Final template

Figure 3.14: Comparison between the result of the first step and second step of the template initialization method for the right globus pallidus bundle.



(a) 2% of fibers from each subject **(b)** Final template

Figure 3.15: Comparison between the result of the first step and second step of the template initialization method for the right putamen bundle.

3.2.3 Conclusions

In this chapter it was shown a method to initialize a bundle template which should reveal all the common features among a set of subject bundles. This is necessary since we are using a gradient descend scheme in the atlas construction. Moreover it preserves the same topology of the bundles which is crucial in the medical research of biomarkers for syndromes like the Gilles de la Tourette's one. In fact, as shown in Chapter 2, we are seeking the main differences between the brain objects of an healthy population and the ones of a population with the GTS syndrome. Therefore we are interested in keeping the information about the path of the fibers and this is possible only maintaining the connectivity between the points. On the other hand, if one uses a sparse representation of the template like in [DFP+09] (see Chapter 1 for more information about sparse representation in the framework of currents), it's no more possible to know where the fibers start and where they end and therefore it becomes impossible to verify the hypothesis linked with the cortico-striato-thalamo-cortical circuits explained in Chapter 2.
Thus it is fundamental to have a template which preserves the connectivity but at the same time it becomes very important also the choice of the number and position of its fibers. We decided to have a template with a number of fibers equal to the average number of fibers of the subject bundles (after the 10% random selection of their fibers). We chose the fibers directly from the subject objects in two passages. The first one was a random selection of 2% of the fibers from each subject object creating a big and dense bundle. In the second step we used a criteria based on the framework of currents to select the "most representative" fibers. In this way we remained also consistent with the rest of the project.

In the next chapter we will show the atlas construction method which means how to update this initial bundle template and how to deform it to each subject bundle in order to find the variability of the template within both populations (controls and patients subject to GTS).

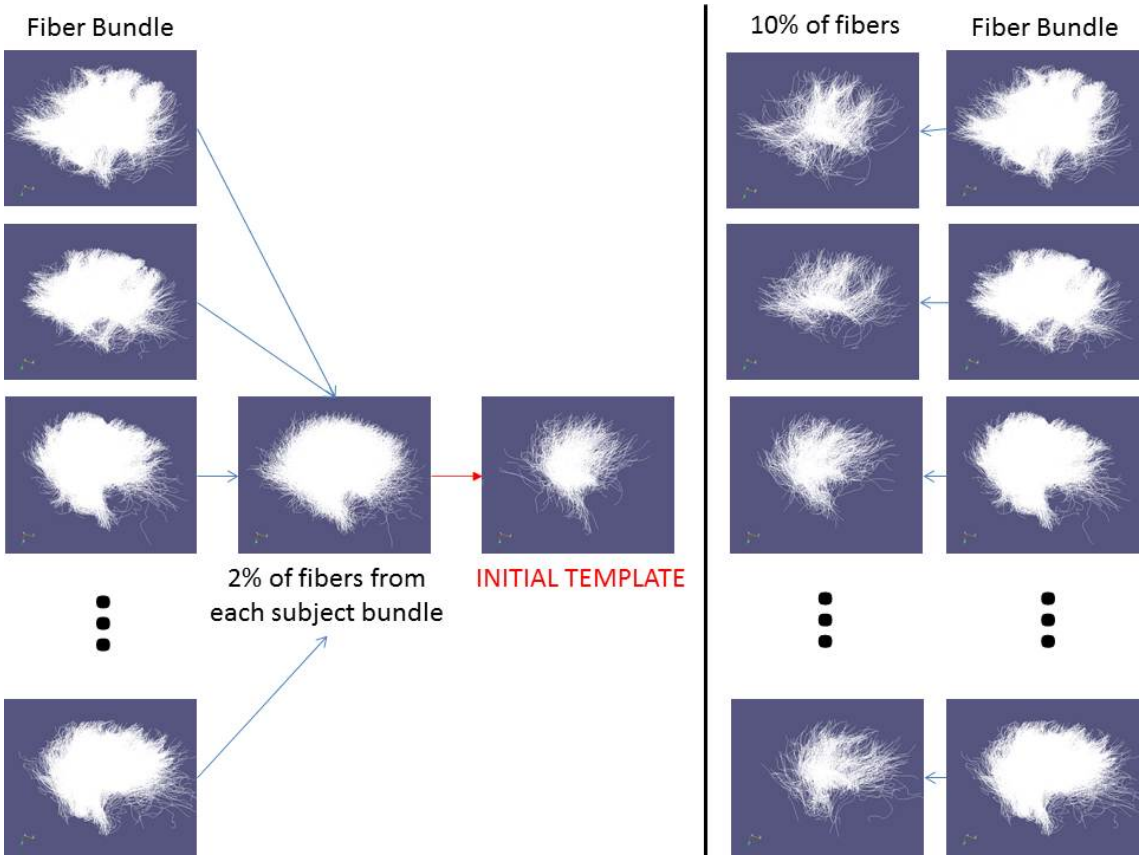


Figure 3.16: Scheme of the bundle template initialization. In the left part there are the results of the two steps of our algorithm. In the right part are shown all the bundles after the 10% random selection of their fibers.

Chapter 4

Atlas construction based on diffeomorphic deformations of currents belonging to the LDDMM framework

Let [$\mathbf{X}^1$,...,$\mathbf{X}^N$] be a set of N shapes belonging to N different subjects, in order to build an *atlas* one needs to:

- Estimate a common average shape called template shape $\mathbf{X}^0$
- Estimate the N template-to-subject deformations ϕ^s from $\mathbf{X}^0$ to [$\mathbf{X}^1$,...,$\mathbf{X}^N$]

In this project, both the template and the subject shapes are meshes or curves and there are 8 curves (fiber bundles) and 9 surfaces (cortical surface and internal structures). It is used a gradient descend scheme to jointly estimate both the template shapes and the deformations. Therefore it is necessary an initial template for each kind of shape. The computation of an initial template for fiber bundles was described in Chapter 3 while we decided to use an ellipsoid for each kind of mesh centred at the mean values of the objects used in the atlas estimation. The objects are also used to create a "bounding box" which gives the dimensions of the ellipsoid.

The goal of this project is to find the differences between the population of controls and the one of patients subject to Gilles de la Tourette syndrome, therefore we need to find a way

of comparing them. In the perspective of Grenander's approach [GM98] the differences between two populations (i.e. healthy and pathological) lie in their variations from a common template representing the features shared by both populations. The variability can be explained using the deformations of the template to each object and hence we need a parametrizable deformation in order to quantify and compare the variations. Moreover, we want to have a kind of parameters which permit to compute statistics on it.

The choice of the deformation framework depends also on the kind of deformation one is dealing with. In this project, since the data are anatomical shapes of the brain, the deformation should capture local non linear variations. The deformation should also be invertible since the flow of information should go back and forth from the template to each object: the template is deformed to the object and looking at the difference between them it can be updated the position of the points/vertices of the template. In addition to that, the deformation framework should also preserve properties such as smoothness of curves or surfaces. Therefore, as suggested in [Dur10], a good choice is diffeomorphism. More exactly, as shown in [Tro95], the Large Diffeomorphic Deformation Metric Mapping (LDDMM) framework seems to be particularly adapted since it can be used for the matching of shapes modeled as currents [VG05] and the metric on its tangent-space uses the same mechanism of RKHS (described in the previous chapters). The tangent-space of a manifold is intuitively a linear approximation to the manifold in a neighbourhood of each point x. This space is given by the linear combinations of the derivatives in all the directions tangent to the manifold. A tangent-space is therefore a vector space where it's possible to compute statistics.

A main characteristic of the atlas estimation used in this project is the utilization of only one diffeomorphism of the underlying space which deforms each template into its relative objects. Every deformation is parametrized by a finite set of elements that we can compare since they come from the same diffeomorphism (more exactly, as we will see later, every deformation share the same set of control points). Moreover, we take into account all the parts of the brains examined in this project and therefore each set of parameters shows the variation of the subject "brain" from the template "brain" estimated. This is also why in the rest of this chapter the word "template" refers to the set of all the different templates which are concatenated together in a single vector in the optimization procedure.

In the first part of this chapter it will be described the kind of deformation used in the atlas construction. Afterwards it will be shown how to match (register) the template into each object and finally how to take into consideration all the objects to estimate the common template.

4.1 Deformation framework: Diffeomorphism

A diffeomorphism is a smooth invertible transformation with smooth inverse. It means that using this kind of transformation connected sets remain connected also after the deformation, disjoint sets remain disjoint and coordinates are transformed consistently [BMTY05]. But of course the inter-subject anatomical variability is likely to involve also topological changes which cannot be captured by diffeomorphism [Dur10]. Thus, the result of a registration (matching) between two shapes will capture only the diffeomorphic variations while the remaining variations will be taken into account in the residuals (difference between the deformed source shape and the target shape).

The main issue of diffeomorphism is its infinite number of degrees of freedom. In fact, differently from a linear transformation which has a finite dimensional parametrization, optimizing a registration criterion over the whole group of diffeomorphism might not be possible [Dur10]. Thus it's necessary to constrain this transformation in a one with a finite number of parameters. In order to do that we use a framework of diffeomorphism constructed via integration of time-varying vector fields which belong to RKHS [GTY04],[JM00].

Let x be the position of a particle and t the time, the integration equation used to define a time-varying deformation is:

$$\phi_t(x) = x + \int_0^t v_s(\phi_s(x))ds \tag{4.1}$$

or equivalently, as flow equation:

$$\begin{cases} \frac{\delta\phi_t(x)}{\delta t} = v_t(\phi_t(x)) \\ \phi_0(x) = x \end{cases} \tag{4.2}$$

where v_t can be seen as the speed vector field in the Eulerian-Lagrangian coordinates and the mapping $t \to \phi_t$ is called flow. Therefore the mapping $x \to \phi_t(x)$ gives the transformation (deformation) of x at time t. The flow starts at time t=0 with ϕ_0 being the identity matrix (id_{R^3} in this project) and it ends at t=1 where $\phi_1 = \phi$, which means that it reaches the desired transformation. In this way one doesn't know only the correspondence between the initial position x and the solution $\phi(x)$, but it is known also the path $\phi_t(x)$. As an example a particle that goes through the position x at time t has velocity $v_t(x)$, and a particle that starts from position x at $t = 0$ will be located at $\phi_1(x)$ at the end of the process (t=1).

This is a clear and easy way of seeing the motion of particles (points in shapes) due to a time-varying vector field acting on the whole space. But it remains to see under which conditions the solution of this process can be considered a diffeomorphism. This is well known from Mechanics and more exactly the mapping $x \to \phi_t(x)$ is considered a diffeomorphism $\forall t \in [0,1]$ if the corresponding time-varying vector field v is a mapping from [0,1] to V, a Hilbert Space of vector fields satisfying certain conditions, such that:

$$\int_0^1 \|v_t\|_V^2 dt < \infty \tag{4.3}$$

then it exists a flow $\phi_t(x)$ which satisfies Eq. 4.1.

V is a Hilbert Space of vector fields with the following restrictions on every element $v \in V$:

1. v is differentiable with continuous differential
2. $v(x)$ tends to zero when $|x| \to \infty$
3. The Jacobian matrix of $v(x)$ $(d_x v)$ tends to zero when $|x| \to \infty$

Moreover we suppose that there is also a constant c_V such that for any $v \in V$:

$$\sup_x (|v(x)| + |d_x v|) \leq \|v\|_V \tag{4.4}$$

This last condition ensures that $\|v\|_\infty \leq c_V \|v\|_V$ and, as explained in [Dur10] and [Gla05], this proves that V is a RKHS. Thus any vector field $v \in V$ can be seen as the speed of a point x at a certain time t and it is the result of the convolution between a squared integrable vector field and a smoothing kernel K^V, also in this case Gaussian, with a a standard deviation equal to λ_V.

4.2 Registration

At each iteration of the atlas estimation the template is deformed into every object. This process is called *registration*. It can be seen as finding the "best" deformation of the space which enables to minimize the dissimilarity between the deformed template and one object. Since the deformed template is also a current, it's possible to use the same

metric as in Chapter 1 and therefore the norm of the difference between the deformed template and an object can be written as:

$$\|(\phi^s(1))_* T - O^s\|^2 \tag{4.5}$$

where the index s refers to one subject. This can be seen as a fidelity-to-data term and the "best" deformation can be found by minimizing it with respect to all the possible deformations ϕ. In fact, as said before, we are looking for one deformation of the entire underlying space which can best match the deformed template into an object. Of course, since the problem might be ill-posed, one needs also a regularity term $Reg(\phi)$. One possibility is to use the total kinetic energy needed to deform the space from identity mapping to the final deformation. In [Gla05] it has been proved and shown that it's possible to pass from the minimization over all possible deformations ϕ to the minimization over all possible time-varying vector fields v (such that Eq.4.3 is still valid). Therefore the minimization process of the registration between the deformed template and an object can be written as:

$$J(v) = \min_v \|(\phi^s(1))_* T - O^s\|^2 + \gamma Reg(v) \tag{4.6}$$

where γ is the trade-off between the fidelity-to-data term and the regularity term. The fidelity-to-data term depends only on the last value ($t = 1$) of the deformation ϕ and not on the whole flow and it can be seen as the distance between the deformed template shape T and the target shape O. Instead the regularization term is defined as:

$$Reg(v) = \int_0^1 \|v_t\|_V^2 dt \tag{4.7}$$

and it depends on the whole span of time $[0, 1]$.

Let G_V be the set of all diffeomorphisms $x \to \phi_t(x)$ for all possible time-varying vector fields $v_t \in$ RKHS described by the kernel K_V, it was demonstrated in [Tro95] that $Reg(v)$ is a distance on G_V and that there is a unique v which achieves the minimum of $Reg(v)$. The associated flow of diffeomorphism ϕ_t is the geodesic path between Id and the resulting deformation ϕ.

Thus the result of the registration process is a time-varying vector field v which depends on the regularization term for the period of time [0,1] and also on the fidelity-to-data term at t=1. Therefore the fidelity-to-data term can be seen as a constrain applied on the final values of the time-varying vector field v.

The flow which minimizes the criterion 4.6 is geodesic in the whole interval [0,1] and, as demonstrated in [Dur10], it is *entirely determined by the initial speed vector field* v_0.

In a discrete setting, the time-varying vector field $v \in$ RKHS is given in the form:

$$v_t(x) = \sum_i K^V(x, c_i(t))\alpha_i^s(t) \quad \text{with } c_i(t) = \phi_t(c_i) \tag{4.8}$$

where $c_i(t)$ are controlled points[1] and x is a generic point. And it is proven in [MTY06] that the geodesic evolution of v_t is such that it has always the form $\sum_i K^V(\cdot, x_i(t))\alpha_i(t)$ at every time step t and such that the momenta $(c_i(t),\alpha_i(t))$ are solutions of the Euler-Lagrange equations:

$$\begin{cases} \frac{dc_i(t)}{dt} = v_t(c_i(t)) = \sum_m K^V(c_i(t), c_m(t))\alpha_m(t) \\ \frac{d\alpha_i^s(t)}{dt} = -\alpha_i^s(t)^T \frac{dv_t}{dc_i} = -\sum_m \alpha_i^s(t)^T \alpha_m^s(t) \nabla_1 K^V(c_i(t), c_m(t)) \end{cases} \tag{4.9}$$

These differential equations can be interpreted as the motion of a certain number of self-interacting particles (the control points) without external forces. The interaction between particles is given by the Gaussian Kernel K^V and also in this case its standard deviation (λ_V) determines the distance between particles which share the same "movement" and deformation. The first equation gives the speed of the control points while the second one explains their accelerations. The speed of any other point x in the space is given by the interpolation with the control points and it is described by Eq.4.8.

As seen before the whole flow of diffeomorphism is entirely determined by the initial values of the time-varying vector field v which are uniquely determined by the values of α_i^s and c_i at t=0, this is called the initial state of the system S_0=$\{\alpha_i(0),c_i(0)\}$. As said before the Regularity term is the total kinetic energy needed to deform the space from identity mapping to the final deformation. But we have already seen that the flow which minimizes the optimization criterion is geodesic and therefore it depends only on the initial values of the time-dependent vector field v. Thus it's possible to use only the initial values of v in the regularization term. Since we are using the RKHS framework, Eq. 4.7 becomes:

$$\|v_0\|_V^2 = \sum_{i=1}^{Nc} \sum_{j=1}^{Nc} \alpha_i^s(0)^T K^V(c_i(0), c_j(0))\alpha_j^s(0) \tag{4.10}$$

[1]Control points are selected initially from a regular lattice and therefore they are not forced to be on the shapes. They are variables of the minimization process and they should move towards the area more variable.

where Nc is the number of control points.

The set of $\{\alpha_i(0)\}$, which are the vectors of the momenta linked with the control points $\{c_i(0)\}$, is exactly the tangent-space representation of the diffeomorphism and it can be seen as a finite dimension parametrization of it. Moreover, being a vector space, one can work on it computing statistics or, for example, finding the directions which explain most of the variability using PCA. This is exactly what it will be shown later.

4.3 Results of the Registration Procedure

Theoretically any "source" shape which is a surface or a curve can be deformed and registered to any other "target" shape with the same technique explained in the previous section. In order to be consistent with the rest of the project we decided to show here the registration process of the initial templates with objects present in both the populations. During the registration process Eq.4.9 is integrated forward in time given initial values of $\alpha_i^s(0)$ and $c_i(0)$. The final values are inserted in Eq.4.8 which is also integrated forward in time for computing the values of the deformed points/vertices of the template. Finally it is computed the gradient of Eq.4.6 with respect to $\alpha_i^s(0)$ and $c_i(0)$ in order to update their values using the gradient descend scheme. When a maximum number of iteration is reached or when the difference between two objective functions of two consecutive iterations is smaller than a tolerance, the algorithm stops.
In the next paragraphs we show the flow of the vertices/points of the template at the final iteration discretized in 20 steps.

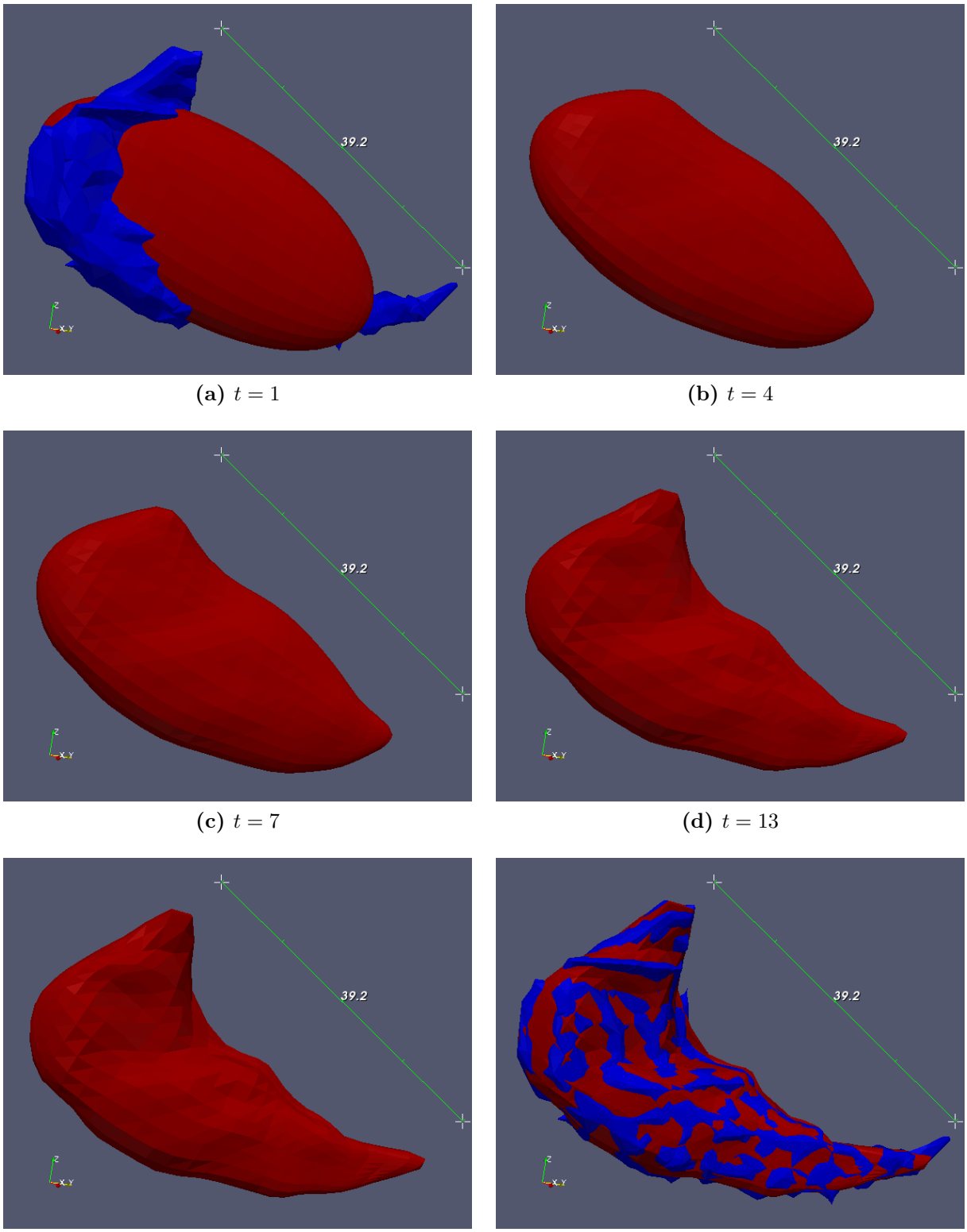
Surfaces/mesh As said before the initial templates of each kind of mesh are ellipsoid. In the next examples we created ellipsoids using 8 subjects and we register them into one of these subjects.

In Fig.4.1 it is shown a flow with parameters: λ_W=2 mm (std Kernel Currents) and λ_V=3 mm (std Kernel Diffeomorphism). An ellipsoid is registered to a left Caudate whose size is about [15mm x 40mm]. 6 different steps of the flow are shown and it's possible to notice that the deformation is really good since the final deformed ellipsoid is quite similar to the object. Only the last part of the horn and other small parts are not well matched. This can be related to the size of λ_W and λ_V but also to the tolerance chosen (the algorithm stops before the maximum number of iterations). λ_W defines the degree of details "seen" by the vector field during the computation of the difference between the template and the object, smaller its value smaller the details seen. λ_V defines the patch of points/vertices of the template which deform in a correlated manner.

In this case, if one wants a better registration, it could be decreased the value of the tolerance or also the values of λ_W and λ_V. But this must be done carefully considering that a lower value of λ_W would permit to the vector field to "see" smaller details but at the same time it reduces the distance "seen" by the vector field between the two objects. Therefore, in this case, the last part of the horn of the object could be too far from the deformed template and the fidelity-to-data term wouldn't participate in the computation of the stepsize in the gradient descend scheme. On the other hand a value too big of λ_W would bring practically to the same situation since any movement of the points/vertices of the deformed template wouldn't change the fidelity-to-data term. From a mathematical point of view one tries to minimize the norm of the difference between the deformed template and the object. If λ_W is too small the two objects are almost orthogonal in the space of currents and therefore their inner product is almost zero. If λ_W is too big the two objects are parallel and their inner product would depend only on their own norms. Therefore the minimization of the norm of the difference would involve only the minimization of the norm of the template without taking into account the target object (since it doesn't change).

On the other hand λ_V is more related to the exactness of the matching and to the number of controls points and relative momenta. Smaller λ_V better the matching between the deformed template and the object (keeping the other parameters constant). λ_V is therefore fundamental since a big value would bring to a too rigid deformation while a value too small could bring to overfitting. In fact, if one wants to analyse the deformations, a "too perfect matching" would detect only features linked to the specific subjects and not to the population. Moreover, the control points are initially positioned in a regular lattice with a distance between them equal to λ_V. This is because the movement of each point/vertex of the space is given by the interpolation with all the control points which depends on the size of the Kernel λ_V (See Eq.4.8). Thus a small value of λ_V brings to a high number of control points.

Other three examples of registration using as "target" objects: left Putamen, left Globus Pallidus and left Thalamus are shown in Fig.4.2. Also in these cases the matching seems quite good but not perfect. In fact, as explained before, we don't want to overfit. This is also really important in the atlas construction since the parameters of the deformations (the set of $\alpha^s(0)$) are used to analyse the differences between the two populations.

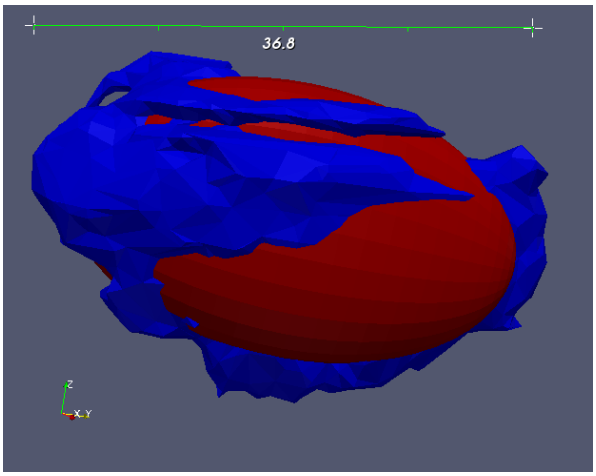


(a) $t = 1$ **(b)** $t = 4$

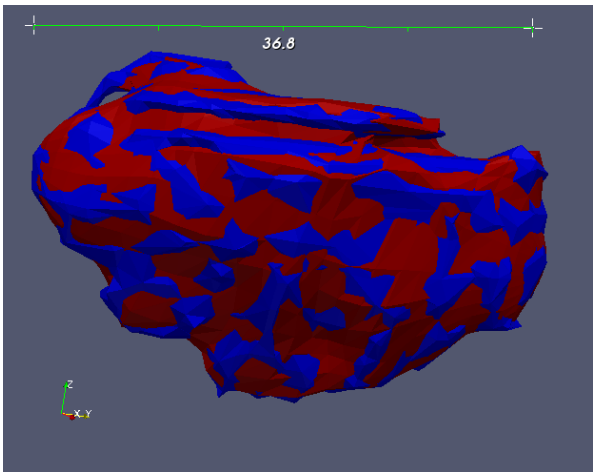
(c) $t = 7$ **(d)** $t = 13$

(e) $t = 15$ **(f)** $t = 20$

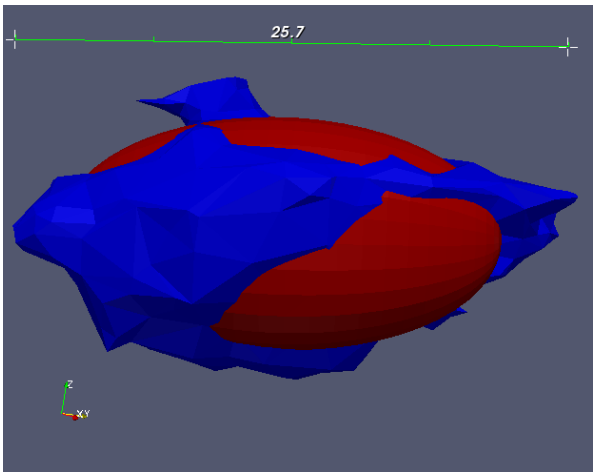
Figure 4.1: Example of registration of an ellipsoid to a left caudate. The diffeomoprhic flow is divided in 20 steps and 6 of them are shown. Parameters: λ_W=2 mm and λ_V=3 mm.



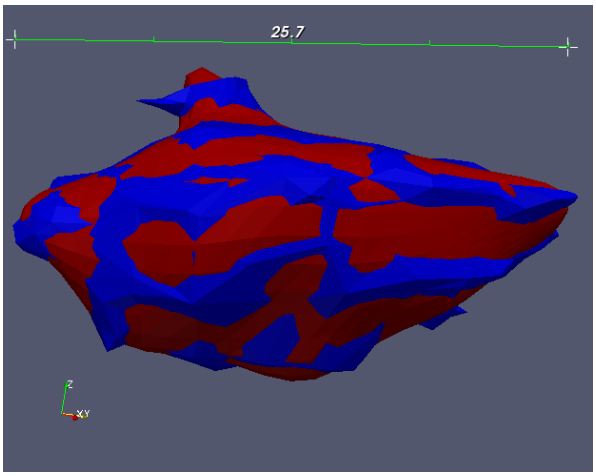
(a) Left Putamen $t = 1$



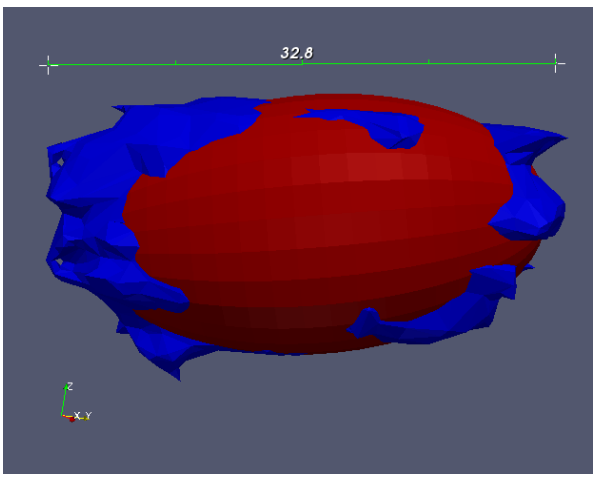
(b) Left Putamen $t = 20$



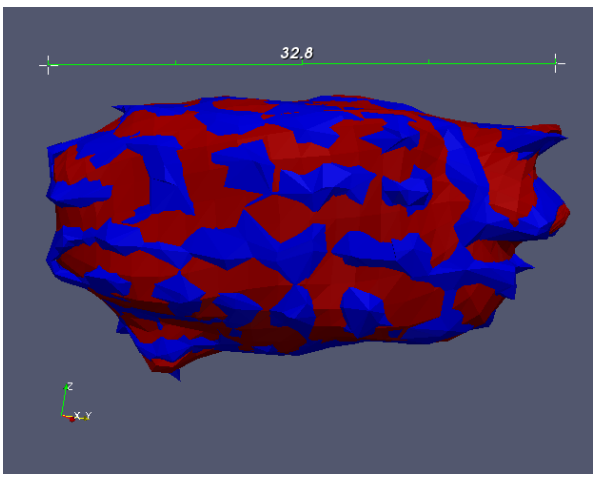
(c) Left Globus Pallidus $t = 1$



(d) Left Globus Pallidus $t = 20$



(e) Left Thalamus $t = 1$



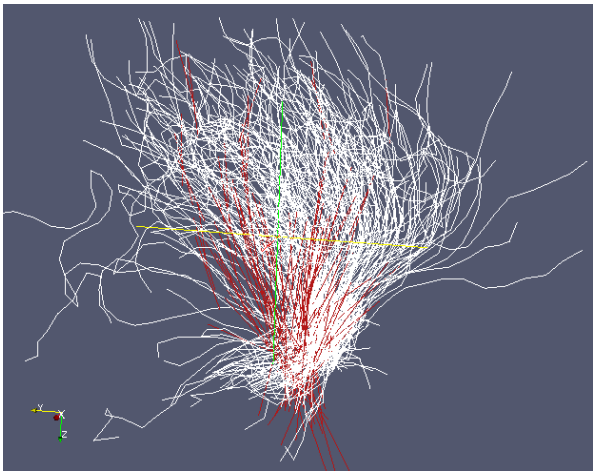
(f) Left Thalamus $t = 20$

Figure 4.2: Example of registration of an ellipsoid to a left Putamen, left Globus Pallidus and left Thalamus respectively. The diffeomoprhic flow is divided in 20 steps. Only the first one and the last one are shown. Parameters: λ_W=2 mm and λ_V=3 mm.

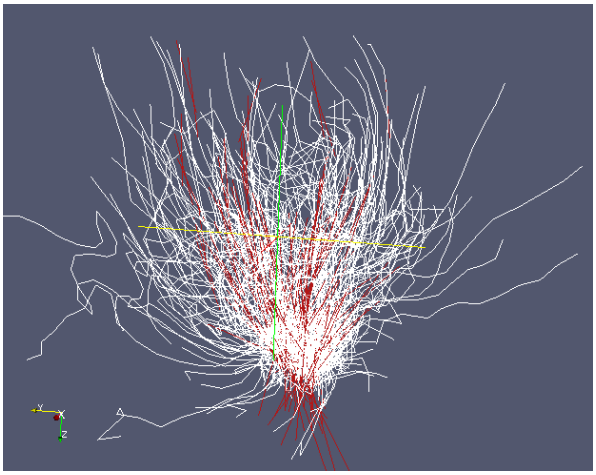
Fiber Bundles The bundle templates used in the following paragraph are the ones obtained with the technique explained in the previous Chapter 3.2.2.

In Fig.4.3 and in Fig.4.4 it is shown respectively the registration between a template of left Globus Pallidus and left Caudate with two different objects approximated by the sparse representation explained in Chapter 1. The results are more complicated to understand than the ones of the surfaces. The high number of fibers doesn't permit to check the exactness of the deformation as easily as with mesh. Nevertheless it seems that the bundle template "shrinks" towards the momenta of the objects. In fact, as explained in Chapter 1, the sparse representation approximates the object with a series of momenta. Thus it seems that the registration process is working also for curves which is one of the new contribution of this project.

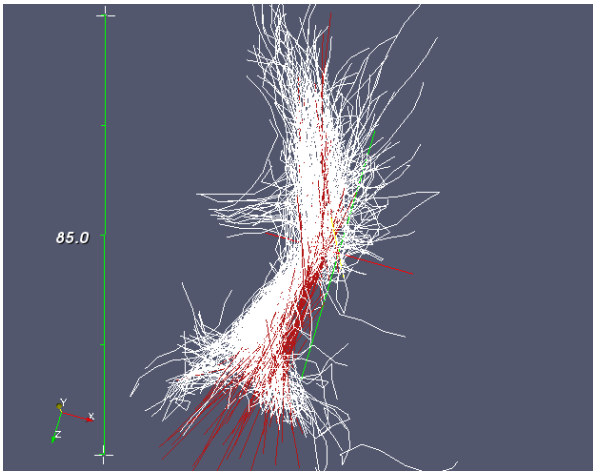
We decided to show a sparse representation as object since this is exactly what we are going to use later to build the atlas. In fact, we decided to use an approximation and not the whole subject's bundle for two main reasons. First the computational time is drastically reduced using the sparse representation since the number of points and momenta is smaller. Second, as described in Chapter 1, a sparse representation makes the object also more interpretable which means that we can choose the degree of precision of the approximation in order to enhance the main "internal structure" of the bundles. This is of course risky since the choice is subjective and it depends on the user but an automatic procedure should also take into account the fact that the bundles are estimated with tractography algorithms which were never shown to produce stable and reproducible results. This is something really complicated that can be seen as a future improvement of this project.



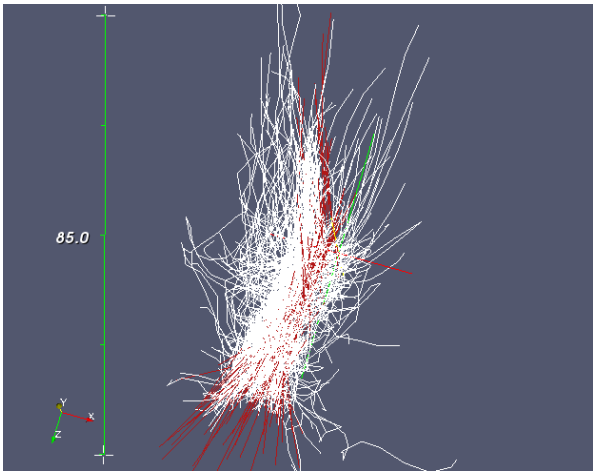
(a) Left Globus Pallidus bundle $t = 1$



(b) Left Globus Pallidus bundle $t = 15$

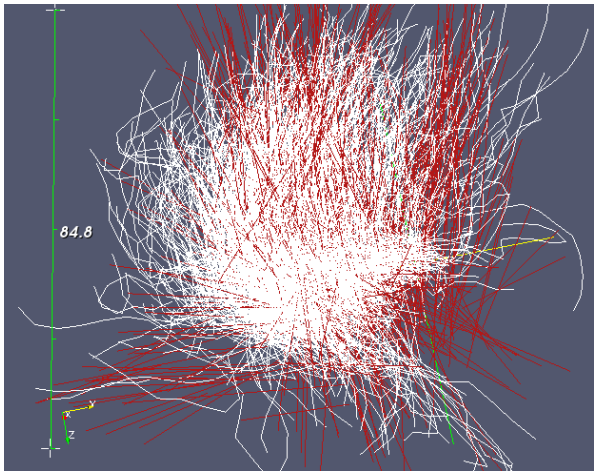


(c) Left Globus Pallidus bundle $t = 1$

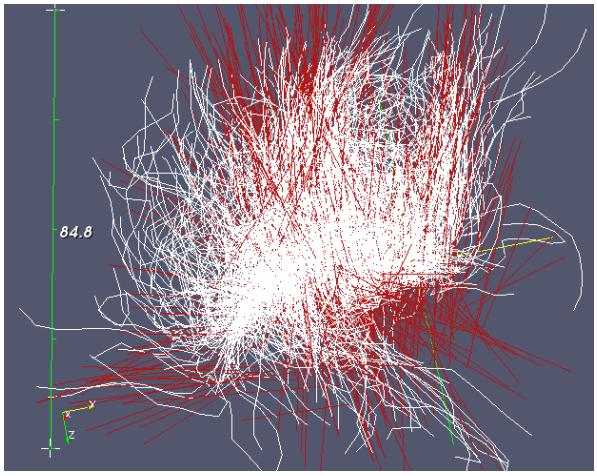


(d) Left Globus Pallidus bundle $t = 15$

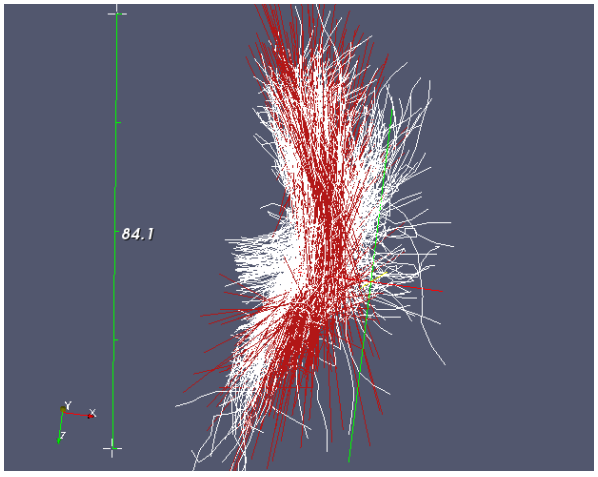
Figure 4.3: Example of registration of a left Globus Pallidus template bundle into a subject's bundle approximated by the sparse representation shown in Chapter 1. The diffeomoprhic flow is divided in 15 steps. Only the first one and the last one are shown. Parameters: λ_W=2 mm and λ_V=10 mm.



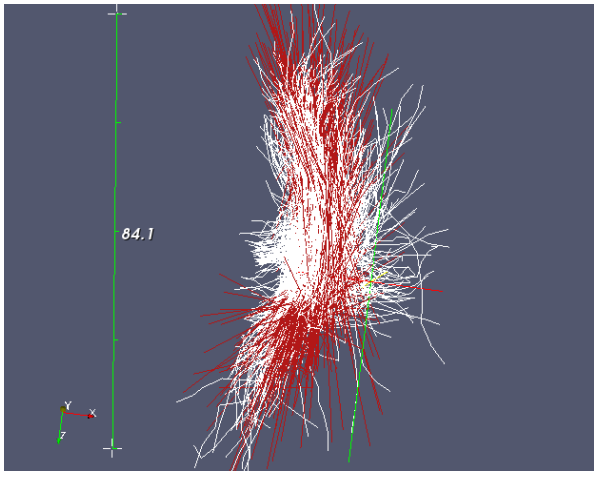
(a) Left Caudate bundle $t = 1$



(b) Left Caudate bundle $t = 15$



(c) Left Caudate bundle $t = 1$



(d) Left Caudate bundle $t = 15$

Figure 4.4: Example of registration of a left Caudate template bundle into a subject's bundle approximated by the sparse representation shown in Chapter 1. The diffeomoprhic flow is divided in 15 steps. Only the first one and the last one are shown. Parameters: λ_W=2 mm and λ_V=10 mm.

4.4 Atlas estimation

In this section it is described how to build an atlas from a set of N subjects which is composed, in this project, by the two populations. The main difference from the previous section is that the template is updated at each iteration in order to become an average shape of the population which captures the common features. Moreover, we chose also to share all the control points among the population leaving only the momenta values as a characteristic of each subject. This is also why we have always used so far a superscript s for the α and not for the c.

Consequently the criterion to minimize becomes:

$$E(\mathbf{X}_0, \mathbf{c}(0), \alpha^{\mathbf{s}}(\mathbf{0})) = \sum_{s}^{N} \left\{ D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s) + \sum_{i=1}^{Nc} \sum_{j=1}^{Nc} \alpha_i^s(0)^T K^V(c_i(0), c_j(0)) \alpha_j^s(0) \right\} \tag{4.11}$$

where $\mathbf{c}(0)$ represents the initial set of control points while $\alpha^{\mathbf{s}}(\mathbf{0})$ is the set of initial α_i for the N subjects and $\mathbf{X}_0$ are the points/vertices of the template. As before the superscript index s refers to the different subjects, instead the subscript index i or j to the different momenta or different control points. All the momenta α_i are vectors starting from the control points. And each control point c_i has N different α_i starting from it, one for each subject. Furthermore, for a clarity purpose, the fidelity-to-data term is indicated with a D.

The main difference from the registration is that now the criterion depends also on the points/vertices of the template and that all the contributions of the subjects are summed together. Moreover now the regularity term can be seen as the kinetic energy needed to deform the template in the N different subject shapes. Or also as the sum of the geodesic distances connecting the template to all the N shapes.

Additionally a L^1 penalty was added to select the most relevant subset of control points [DPK+12]:

$$\sum_{i=1}^{Nc} \|\alpha_i^s(0)\|_1 \tag{4.12}$$

Thus the complete criterion can be written as:

$$E(\mathbf{X}_0, \mathbf{c}(0), \alpha^{\mathbf{s}}(\mathbf{0})) = \sum_s^N \left\{ \frac{1}{2\sigma^2} D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s) + \sum_{i=1}^{Nc}\sum_{j=1}^{Nc} \alpha_i^s(0)^T K^V(c_i(0), c_j(0))\alpha_j^s(0) + \gamma \sum_{i=1}^{Nc} \|\alpha_i^s(0)\|_1 \right\} \tag{4.13}$$

The two values σ^2 and γ balance the fidelity-to-data term and the L^1 term against the regularity term. Eventually the only variables to be optimized are:

1. $\mathbf{X}_0$: the position of the vertices (for surfaces) or of the points (for curves) of the template.
2. $\mathbf{c}(0)$: the position of the control points which are in the same domain of the template.
3. $\boldsymbol{\alpha^s}(0)$: the set of initial momenta that parametrizes each template-to-subject deformation

Using as optimization method a gradient descend scheme, it is easy to notice that only the first two quadratic terms are differentiable. The differentiation criterion can be done independently for each subject s, and while for the regularity therm it is really easy to compute the derivatives with respect to α_i and to c_i, it can not be said the same for the fidelity-to-data term. In fact the differentiation of the L^2 regularity term brings to:

$$\frac{\partial \left\{ \sum_{i=1}^{Nc}\sum_{j=1}^{Nc} \alpha_i^s(0)^T K^V(c_i(0), c_j(0))\alpha_j^s(0) \right\}}{\partial \alpha_i} = 2\sum_{j=1}^{Nc} K^W(c_i, c_j)\alpha_j^s \tag{4.14}$$

$$\frac{\partial \left\{ \sum_{i=1}^{Nc}\sum_{j=1}^{Nc} \alpha_i^s(0)^T K^V(c_i(0), c_j(0))\alpha_j^s(0) \right\}}{\partial c_i} = 2\sum_{j=1}^{Nc} (\alpha_j^s)^T \alpha_i^s \nabla_1 K^W(c_i, c_j) \tag{4.15}$$

where, since in this project we are using a Gaussian Kernel which is symmetric, scalar and isotropic of the form : $K^W(x, y) = \exp(-\|x - y\|^2/\lambda_W^2)\mathrm{Id}$, the value $\nabla_1 K^W(c_i, c_j)$ is equal to $\frac{\partial K^W(c_i, c_j)}{\partial c_i} = -\frac{2}{\lambda^2}(c_i - c_j)K^W(c_i, c_j)$.

Instead the computation of:

$$\begin{cases} \nabla_{\alpha_i} D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s) \\ \nabla_{c_i} D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s) \\ \nabla_{\mathbf{X}_0} D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s) \end{cases} \tag{4.16}$$

is really much more difficult and it can be found in [DPK+12]. Practically the idea behind it is to integrate forward in time the flow equations 4.9 and 4.8 where x are the vertices or points of the template. In this way one finds the final deformations $\phi^s(1)(\mathbf{X}_0)$ of the template which is used to compute the gradient of the fidelity-to-data term with respect to the points of the template at time t=1: $\nabla_{\mathbf{X}_0(1)} D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s)$ as shown in [VG05]. This indicates the direction to move the vertices/points of the template in order to decrease the most the residual current. This information is consequently propagated to the control points and momenta integrating backward from $t = 1$ to $t = 0$ two ODEs shown in [DPK+12]. The results are exactly the gradients of $D(\phi^s(1)(\mathbf{X}_0), \mathbf{X}^s)$ with respect to the momenta and control points which are used to update the initial positions of control points $\mathbf{c}(0)$ and the initial momenta $\boldsymbol{\alpha}^s(0)$. Another ODE is afterwards integrated backward from $t = 1$ to $t = 0$ giving as result the gradient of the fidelity-to-data term with respect to the vertices/points of the template (at $t = 0$) which is used in the gradient descend scheme to update the initial positions of the vertices/points of the template. Control points and template points/vertices are shared among all the subjects and thus all the contributions from each subject are summed together. Whereas the momenta α are a characteristic of each subject.

At this point one can compute the values of the template points/vertices and control points summing the results from the fidelity-to-data term and regularization term. But, in order to find the momenta $\boldsymbol{\alpha}^s$ values, one needs also to take into account the L^1 penalty. In order to do that, as suggested in [DPK+12], it is used an adapted gradient-descent scheme called Fast Iterative Shrinkage and Thresholding Algorithm (FISTA) [BT09], [YGSM10].

This algorithm belongs to the class of Proximal Gradient Methods and it is a faster version of the well known method ISTA. Let F(x) be a cost function to minimize and Q(x,y) a quadratic approximation of F(x) around a well chosen point y, the goal of this kind of methods is to minimize Q(x,y) instead than F(x).

As an example, let F(x) be a composite objective function given by:

- f : $\mathbf{R}^3 \rightarrow \mathbf{R}$ a smooth convex function of the type $C^{1,1}$ which means continuously differentiable with a Lipschitz continuous gradient:

$$\|\nabla f(x) - \nabla f(y)\| \leq L(f)\ \|x-y\|\ \forall\ x,y \in \mathbf{R}^3$$

where L(f) is always greater than zero and it's called the Lipschitz constant of ∇f.

- g : $\mathbf{R}^3 \to \mathbf{R}$ a continuous convex function which is possibly *non smooth*.

Thus, F(x) can be written as:

$$F(x) = f(x) + \lambda g(x) \tag{4.17}$$

and in this project f(x) is the fidelity-to-data term with the regularity term while g(x) is the L^1 penalty which is therefore separable and it can be written as:

$$g(x) = \sum_i g_i(x_i) = \sum_i |x_i| \tag{4.18}$$

From now on we will consider x as the only variable of this minimization process which can be seen as the set of momenta $\boldsymbol{\alpha}^s$ used in this project. In fact, both the vertices/points of the template and the control points don't depend on the L^1 penalty and they can be considered already updated and therefore they are not variables of the FISTA process. Thus x is a set of vectors belonging to $\mathbf{R}^3$.

In order to minimize Eq.4.17 it is used, as in ISTA, an updating rule based on a second-order approximation of $f(x)$ where the Hessian is diagonalized by: $L\mathrm{Id}$ and L is a positive Lipschitz constant which plays the role of stepsize:

$$\begin{aligned} x_{k+1} &= \min_x f(x_k) + (x - x_k)^T \nabla f(x_k) + \frac{1}{2}\|x - x_k\|_2^2 \nabla^2 f(x_k) + \lambda g(x) \\ &\sim \min_x (x - x_k)^T \nabla f(x_k) + \frac{L_k}{2}\|x - x_k\|_2^2 + \lambda g(x) \\ &= \min_x \frac{1}{2}\|x - u_k\|_2^2 + \frac{\lambda}{L_k}\sum_i g_i(x_i) \\ &= \mathrm{p}_L(x_k) \end{aligned} \tag{4.19}$$

where

$$u_k = x_k - \frac{1}{L_k}\nabla f(x_k) \tag{4.20}$$

The constant term $f(x_k)$ was subtracted in the first passage and the two constant terms $-\frac{(\nabla f(x_k))^T x_k}{2L_k}$ and $\frac{(\nabla f(x_k))^T(\nabla f(x_k))}{2(L_k)^2}$ were added in the last passage since they don't change the result of the minimization (i.e. differentiation with respect to the variable x, which is not present in these constant terms). Moreover in the last passage $f(x)$ and $g(x)$ are divided by the constant L_k, since also this computation doesn't change the result.

Now, it's easy to see that this cost function has a closed form solution with respect to each x_i^s where the superindex s refers to the different subjects and the subindex i to each momenta. In the following the subindex k is used to indicate the different iterations:

$$\begin{aligned} x^s_{i,(k+1)} &= \min_{x_i^s} \frac{(x_i^s - u^s_{i,k})^2}{2} + \frac{\lambda |x_i^s|}{L_k} \\ &= \text{soft}\left(u^s_{i,k}, \frac{\lambda}{L_k}\right) \end{aligned} \tag{4.21}$$

where

$$\begin{aligned} \text{soft}(u, L) &= \text{sgn}(u)\max(|u| - L, 0) \\ &= \begin{cases} \text{sgn}(u)(|u| - L) & \text{if} |u| > L \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4.22}$$

is called *soft-thresholding* or *shrinkage* function [YGSM10].

This algorithm is called ISTA and it has a worst-case complexity result of $O(\frac{1}{k})$. The FISTA algorithm used in this project is faster with an improved complexity result of $O(\frac{1}{k^2})$ [BT09]. The main difference with the previous algorithm is that x at iteration $k+1$ is not found using x_k but another point y_k which is a very specific linear combination of the previous two points (x_k,x_{k-1}).

This idea behind FISTA was firstly introduced by the Russian Nesterov [Nes83] in 1983 for minimizing a *smooth* convex function. In the paper of Beck et al. [BT09] the problem is instead convex but *nonsmooth* due to the L^1 penalty (exactly as in this project).

Also in FISTA, Eq.4.17 is minimized by a quadratic approximation Q(x,y) which takes exactly the same form as before:

$$\min_x Q(x, y) = \text{soft}(u_k, \frac{\lambda}{L_k}) \tag{4.23}$$

where $u_k = y_k - \frac{1}{L_k}\nabla f(y_k)$. But in this case y_k is not equal to x_k but it is:

$$y_k = x_k + \frac{t_{k-1} - 1}{t_k}(x_k - x_{k-1}) \tag{4.24}$$

where t_k is a positive real sequence satisfying $t_k^2 - t_k \leq t_{k-1}^2$.

A possible drawback of this scheme is that the Lipscitz constant L (the stepsize) can be hard to compute. Therefore it can be used a backtracking line search strategy to generate a scalar sequence of step sizes that approximates L_k. Given a constant value η greater than 1 and y_k, it's possible to define $L_k = \eta^j L_{k-1}$, where j is the smallest nonnegative integer such that the following inequality holds:

$$F(\mathrm{p}_L(y_k)) \leq Q(\mathrm{p}_L(y_k), y_k) \tag{4.25}$$

where $\mathrm{p}_L(y_k)$ is the result found in Eq.4.19, and it means that one needs to find the smallest L_k for which the approximation is still valid (i.e. since Q(x,y) is always greater than F(x), the minimum of Q(x,y) will be the smallest value of F(x) using that approximation). Then, once the stepsize is computed, one can apply the FISTA method using the following three equations:

$$t_k = \frac{1 + \sqrt{4t_{k-1}^2 + 1}}{2} \tag{4.26}$$

$$y_k = x_k + \frac{t_{k-1} - 1}{t_k}(x_k - x_{k-1}) \tag{4.27}$$

$$x_{k+1} = \mathrm{soft}(y_k - \frac{1}{L_k}\nabla f(y_k), \frac{\lambda}{L_k}) \tag{4.28}$$

The specific Eq.4.26 emerges during the proof of the gain in convergence rate from $O(\frac{1}{k})$ to $O(\frac{1}{k^2})$ in Nesterov's paper [Nes83].

As said before, the f(x) function used so far can be substituted with the fidelity-to-data term and the regularity term while the function g(x) is the L^1 penalty term. Thus, calling $\nabla_{\alpha_i^s(0)} f$ the gradient of f with respect to the i^{th} α at time 0 and belonging to subject s, it's possible to write a sketch of the algorithm applied to this project, where $\alpha_{i,k}^s(0)$ means the value of $\alpha_i^s(0)$ at the k^{th} iteration. The soft function ensures sparsity since it zeros out the values of $\alpha_i^s(0)$ which are too small in magnitude [DPK+12].

Algorithm FISTA

Input: f: $\mathbf{R}^3 \rightarrow \mathbf{R}$

0: **for** each subject s
1: **for** each momenta α_i^s
2: Set $\alpha_{i,0}^s$=0, $\alpha_{i,1}^s$=0, t_0=1, t_1=1, k=1
3: Initialize L_0>0, η>1, λ>0
4: **while not converged do**
5: $y_{i,k}^s = \alpha_{i,k}^s + \frac{t_{k-1}-1}{t_k}(\alpha_{i,k}^s - \alpha_{i,(k-1)}^s)$
6: if (k>1) Find the smallest nonnegative integer i, such that L=$\eta^i L_{k-1}$ and the condition $F(\mathrm{p}_L(y_k)) \leq Q(\mathrm{p}_L(y_k), y_k)$ is respected. Then, set L_k=$\eta^i L_{k-1}$.
7: $u_{i,k}^s = y_{i,k}^s - \frac{1}{L_k}\nabla f(y_{i,k}^s)$
8: $\alpha_{i,(k+1)}^s = \mathrm{soft}(u_{i,k}^s, \frac{\lambda}{L_k})$
9: $t_{k+1} = \frac{1+\sqrt{4t_k^2+1}}{2}$
10: k = k+1;
11: **end while**
Output $\alpha_i^s(0)$=$\alpha_{i,k}^s$

Table 4.1: Sketch of the FISTA Algorithm applied to this project

4.5 Considerations

In this section it has been shown how to build an atlas preserving the topology of the template throughout the construction. In fact, the template points/vertices are moved preserving the connectivity between them. Moreover the template and the control points are shared among all the subjects and therefore they show the anatomical features in common. Instead the momenta $\boldsymbol{\alpha}(0)$ are unique for each subject and, as explained before, they can be used to perform statistical analysis such as PCA. Furthermore the sparsity prior permits to have a compact representation of shape variability within the population which has great potential for clinical studies since it helps the interpretation of the results. In conclusion it's useful to recapitulate the 4 parameters set manually by the user and used throughout the atlas construction:

1. λ_W : Kernel size used in the currents framework. It's fundamental in the fidelity-to-data term since it describes the area of influence of the vector field w around the shapes. It means that, using for example two curves, only when they are inside the area of influence of the other, a movement of one of them is "seen" by the vector field w and it influences the fidelity-to-data term. But if the curves are further from each other than their area of influence, any variation of one curve is not "seen" by the vector field w and therefore the fidelity-to-data term remains constant. From a mathematical point of view when λ_W is too small the two curves are almost orthogonal in the space of currents and therefore their inner product is almost zero. On the other hand if λ_W is too big the two curves are parallel in the space of currents and their inner product depends only on their own norms. Therefore the minimization of the fidelity-to-data term involves only the minimization of the norm of the template without taking into account the target object since it doesn't change.
   This behaviour brings to two considerations. When the two curves are far away from each other, one needs a big value of λ_W to compare them. But in this way it's impossible to check the small details of every curve since the vector field is too smoothed and it can't interpolate perfectly all the points between the two curves. On the other way, if the two curves are superimposed (as in this project) we can use also a small value of λ_W which enables us to compare the small variations of each curve. Therefore, the degree of detail of the anatomical features one wants to detect is related also to the distance between the two curves. This also why we decided to use a rigid registration at the beginning of the atlas estimation: reducing the distances between the objects gave us the possibility to use any λ_W and its choice was only related to the anatomical detail we wanted to take into account.
2. λ_V : Kernel size used in the diffeomorphic deformation framework. This parameter controls the "rigidity" of the diffeomorphism which means that bigger is this parameters more points move in a correlated manner during the deformation. Thus,

this parameter controls the type of deformation one wants to use: a value towards zero means that each point can move in an autonomous way from the rest (many degrees of freedom) while a value towards infinite means that every point will move in the same way as the others creating an almost rigid transformation (few degrees of freedom). That's why this value is important to set up the quality of deformation one wants to achieve. Moreover when the value of this parameter is too low the registration between the template and each object can be "too perfect" bringing to overfitting. This is typical of models too complex since they adapt too much to the data and an analysis of the deformations might not detect features characterizing the whole population but only the few subjects under examination. Thus it's important to find a good compromise between the exactness of the registration and the smoothness of the time-varying vector field v. This is also linked to the number of control points one wants to have. In fact, Eq.4.8 describes the deformation of each point in the template shape which depends on the interpolation with the control points and their momenta using the Gaussian Kernel K^V. λ_V defines the maximum distance that a control point can have in order to modify the template point/vertex. Thus, smaller λ_V more the control points and longer the computational time of the whole atlas construction.

3. σ^2 : it's the trade off value between the fidelity-to-data term and the regularity term. Bigger this value, greater the importance of the regularity term and therefore smaller the norm of the vector field v. This is extremely important in the Atlas construction since it balances the position of the updated common template. A value too small brings to a good matching of the template to each object but it prevents the updated template to move towards the center of the population. A good trade-off is therefore fundamental in the Atlas construction and its choice should be automatic. This can be seen as a future improvement of this project.
4. γ : it's the parameter linked with the L^1 penalty. Bigger this value, greater the importance of the L^1 term which gives a sparser solution and therefore less control points with an α_i^s value different from zero. It's the trade-off between the number of degrees of freedom and the variability captured by the atlas. Also in this case the choice of the value of the parameter should be automatic.

4.6 Results of the Atlas Construction

In this section we show the results of the Atlas construction procedure. We focus mainly on the difference with respect to the Registration process, which means the update of the template using the contribution of all the subjects under examination.

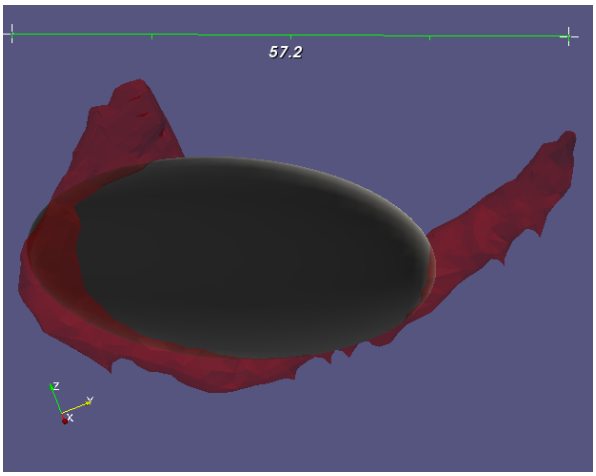
Two independent Atlas Construction are shown in Fig.4.5 and in Fig.4.6 respectively.
In the first one 4 kind of brain structures are used (left Caudate, Putamen, Globus Pal-

lidus and Thalamus) with 8 subjects (4 controls and 4 patients). The parameters are: λ_W=2 mm for each object and λ_W=5 mm for the diffeomorphism. The final number of momenta is 636.
The second Atlas Construction regards only two kind of bundles: left Caudate and left Globus Pallidus. Also in this case we use 4 controls and 4 patients. The parameters are:λ_W=2 mm for each bundle and λ_W=10 mm for the diffeomorphism. The final number of momenta is 819.

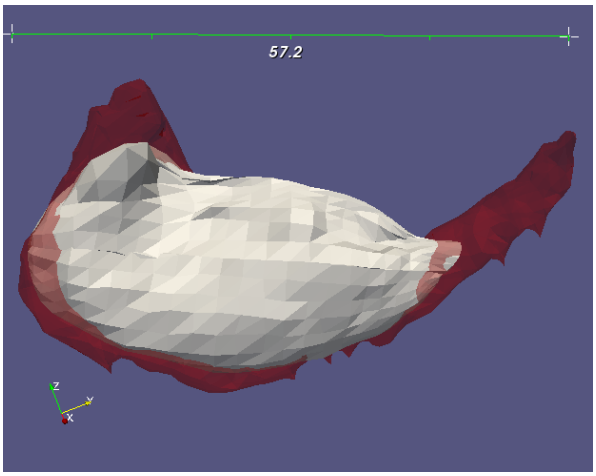
Also in this case it's easier to notice the change in the mesh initial template than in the bundle one. This is also related to the fact that we begin the atlas construction in the mesh case with an ellipsoid which is quite different from the objects used in the Atlas Construction while in the bundle case the template is already a "common shape" and therefore the change is smaller than in the mesh case. Moreover the shape of the bundle is not a closed surface as in the mesh case but it is constituted by different "open" fibers which make the visual inspection of the results more difficult. Nevertheless it's possible to notice some changes looking at the initial and final template of the two bundles superimposed in Fig.4.6. Theoretically this change should take into account the information from all the subjects in order to show the common features.

After that, we decide to show also an example of sparse representation used in the second Atlas construction and the relative registration of the final template (Fig.4.7). It's possible to notice how the sparse representations are quite different among them and the final template moves a little bit towards the center of this small population deforming also its fibers in order to become an "average shape" of this particular set of subjects.

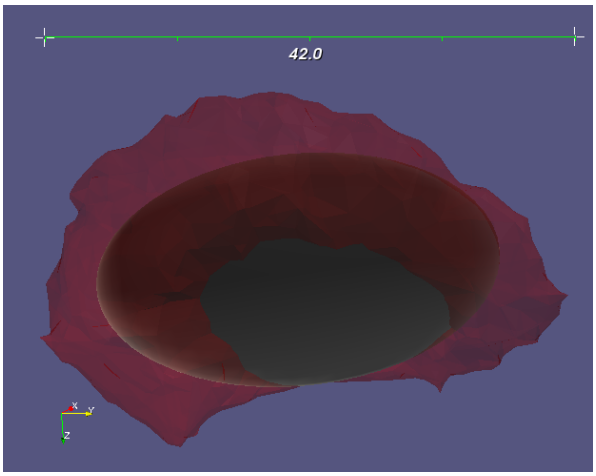
The choice of the parameters is quite difficult especially for fiber bundles. The value of λ_W used in the sparse representation is also used in the Atlas Construction since it sets the "resolution" of the fiber bundle seen as a current. This value is set manually by the user. Of course, an automatized general procedure would be better but this would depend on, for example, the tractography method used, the degree of redundancy one wants to keep and the kind of data analysed. This is out of the scope of this Master's Thesis and it can be seen as a future improvement.



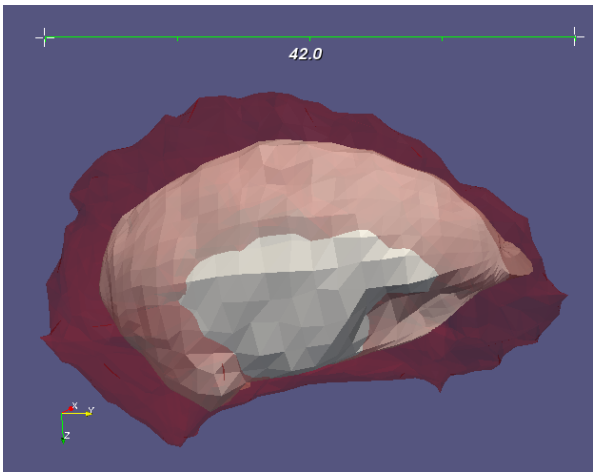
(a) Initial template of left Caudate



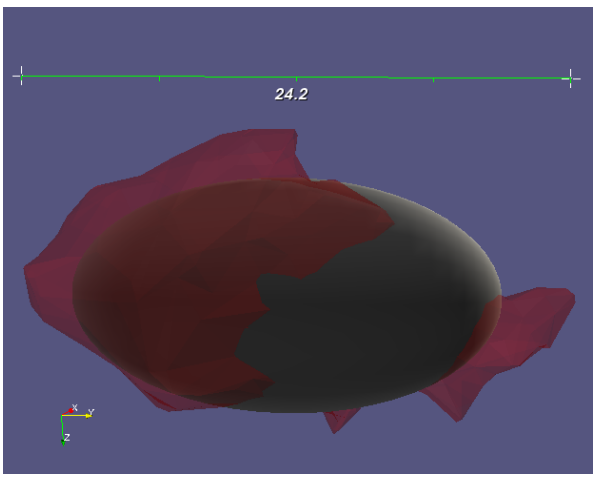
(b) Last template of left Caudate



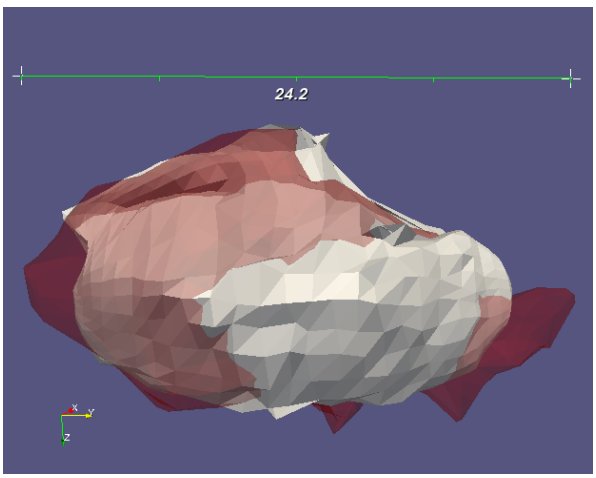
(c) Initial template of left Putamen



(d) Last template of left Putamen

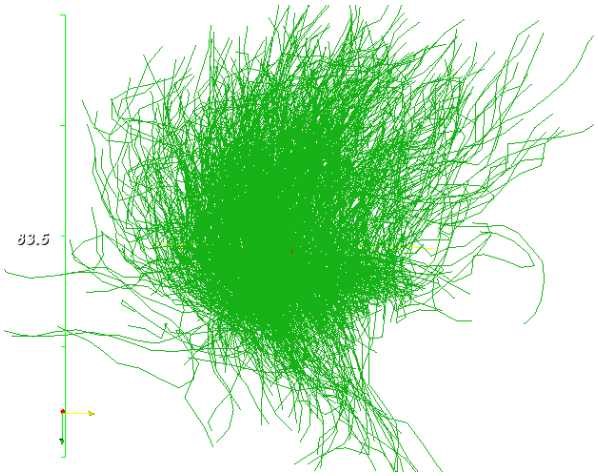


(e) Initial template of left Globus Pallidus

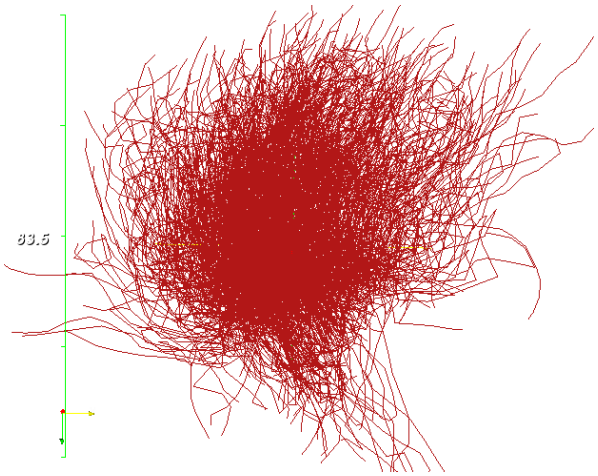


(f) Last template of left Globus Pallidus

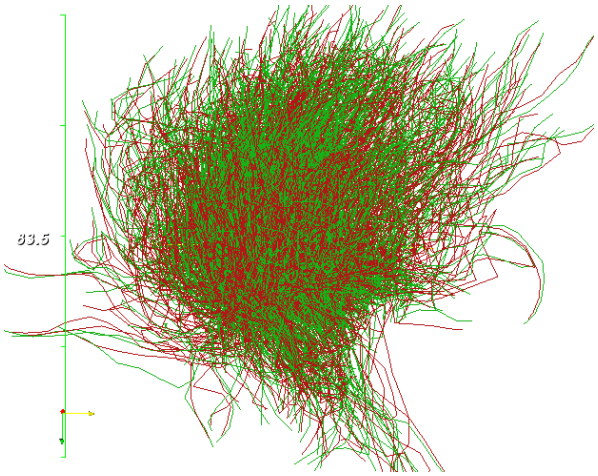
Figure 4.5: Initial and last template of three internal structures: left Caudate, left Putamen and left Globus Pallidus. The Atlas Construction procedure takes into account 8 subjects (4 controls and 4 patients). Parameters used are: λ_W=2 mm for each object and λ_W=5 mm for the diffeomorphism. The object in the background with a greater opacity is one of the objects used in the Atlas construction.



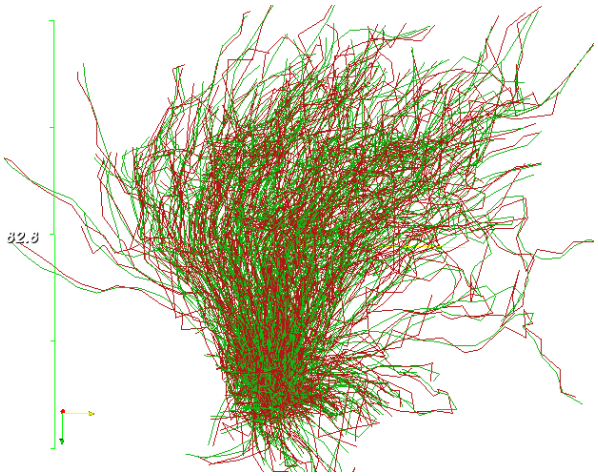
(a) Initial template of left Caudate bundle



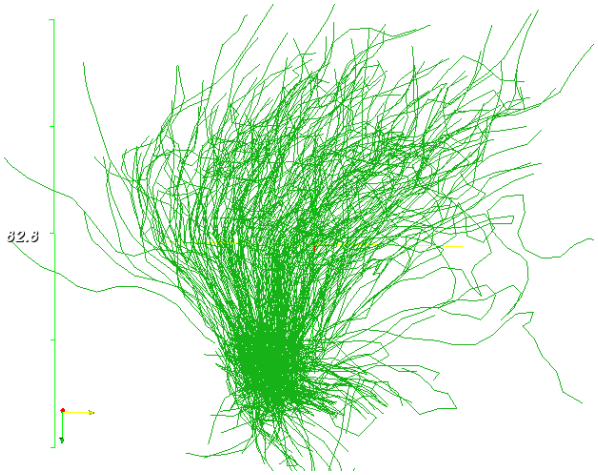
(b) Final template of left Caudate bundle



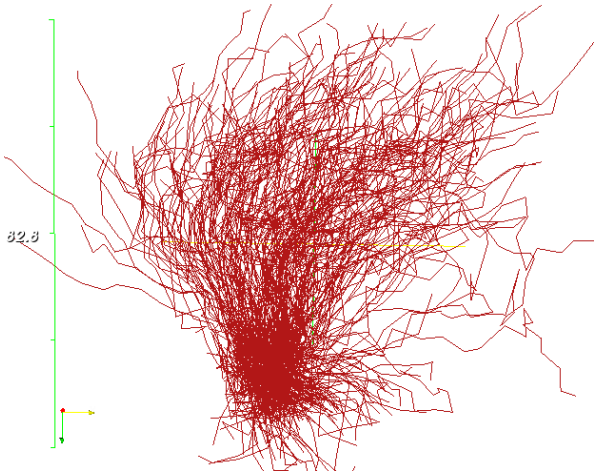
(c) Initial and last template of left Caudate bundle together



(d) Initial and final template of left Globus Pallidus bundle together

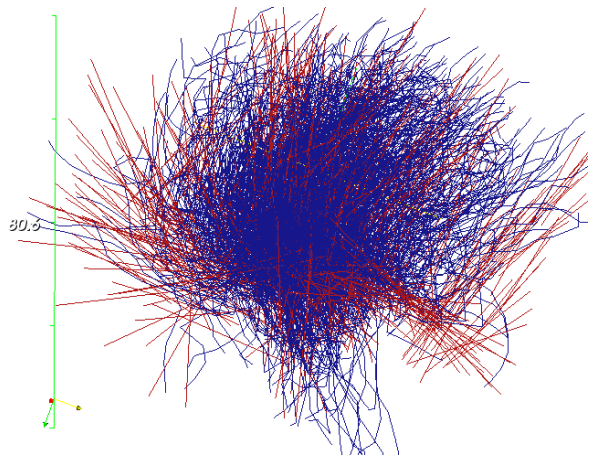


(e) Initial template of left Globus Pallidus bundle

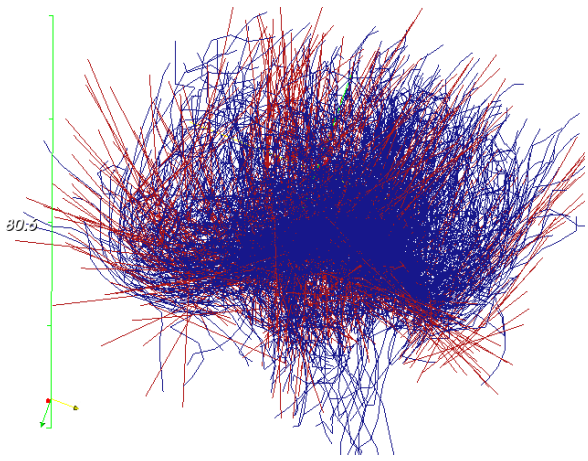


(f) Final template of left Globus Pallidus bundle

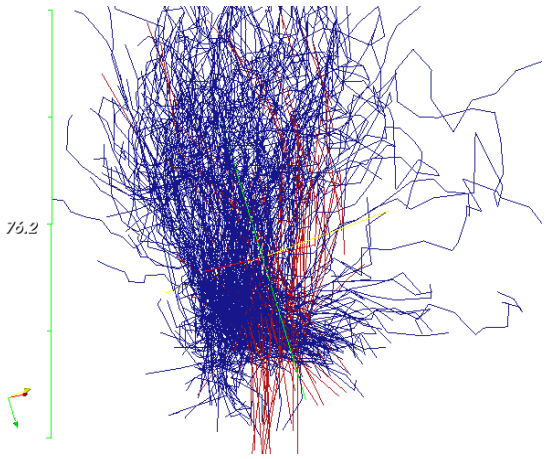
Figure 4.6: Initial and last template of left Caudate bundle and left Globus Pallidus bundle. The Atlas Construction procedure takes into account 8 subjects (4 controls and 4 patients). Parameters used are: λ_W=2 mm for each bundle and λ_W=10 mm for the diffeomorphism.



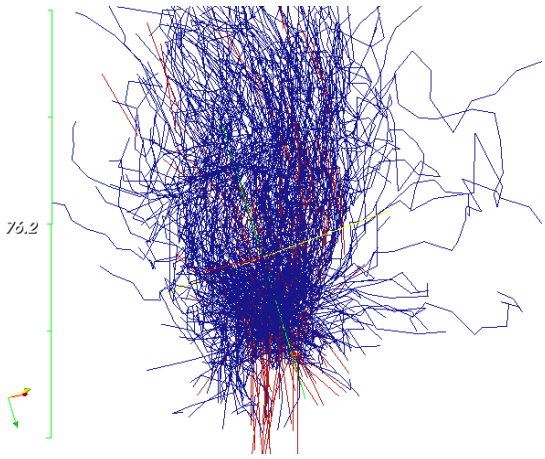
(a) Final template of left Caudate bundle



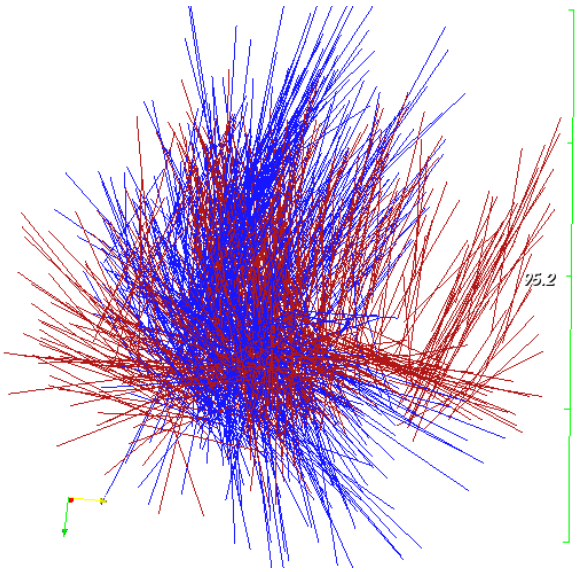
(b) Deformation of final template of l.C. bundle



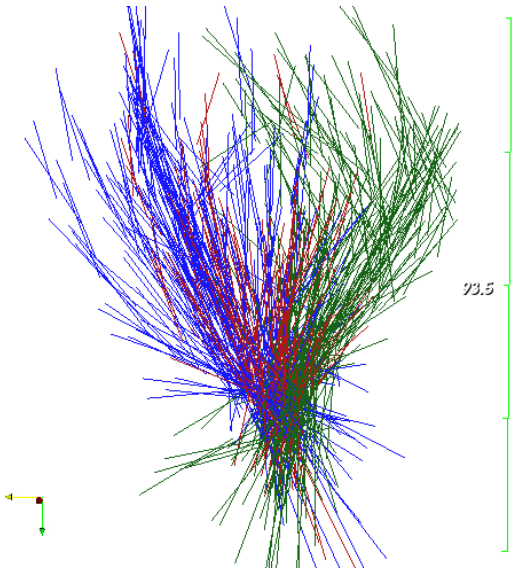
(c) Final template of left Globus Pallidus bundle



(d) Deformation of final template of l.G.P. bundle



(e) 2 sparse representations of left Caudate bundle



(f) 3 sparse representations of left G.P. bundle

Figure 4.7: Example of deformation of the final template into a subject's bundle. The Atlas Construction is based on 8 subjects (4 controls and 4 patients) with λ_W=2 mm for each bundle and λ_W=10 mm for the diffeomorphism. The sparse representation is based on a λ_W=2 mm with τ=15% of std.

4.7 Analysis of the results

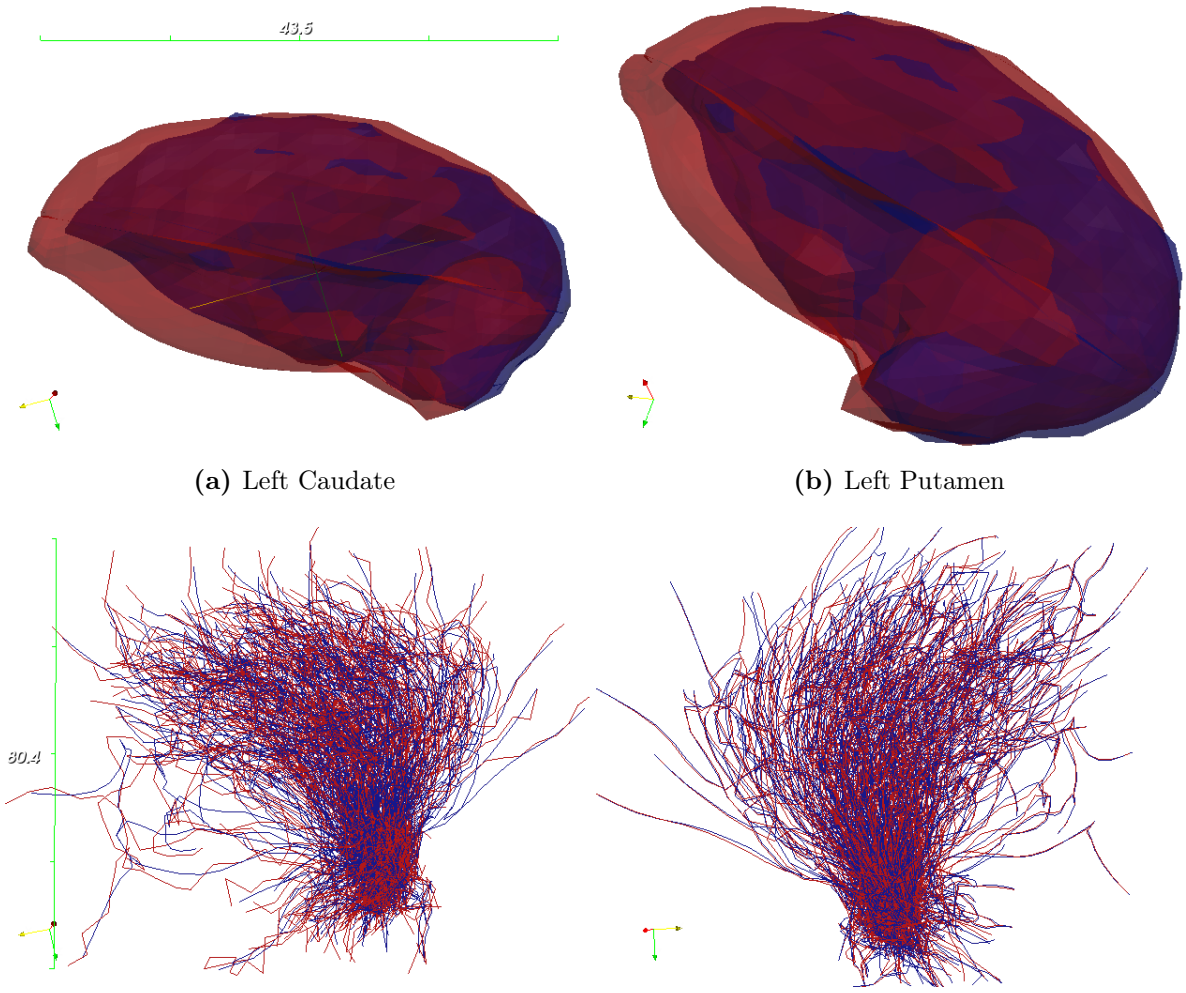
There are several ways to analyse the results of the Atlas Construction procedure. In the following we show three of them:

1. Visual Inspection of the templates of the two populations resulting from two independent Atlas Constructions.
2. Visual Inspection of the "mean deformation" of only controls or patients starting from the same common template.
3. Principal Component Analysis (PCA) of the diffeomorphic deformations of an Atlas built using both the populations.

Template Comparison The first technique is about comparing the final templates of two different and independent Atlas Constructions using in the first one only controls and in the second one only subjects. In this way it's possible to compare the common features of both the populations. A visual inspection is easy to understand and really practical for sharing information with people from other research area such as doctors or neurologists.

In Fig.4.8 are shown two different examples of template comparison. The first two figures refer to the template mesh of the left Caudate and left Putamen while the last two are about the templates of the left Globus Pallidus bundle. In these three cases we used 4 controls (red) and 4 patients (blue). Unfortunately 4 subjects don't represent a significant population and therefore we can't draw any conclusion from these images. But it's possible to understand how this method could be very useful if employed with bigger populations. In fact, if these figures had been taken from a bigger population, one could have observed that the Left Caudate template is definitively longer in the population of patients than in the one of controls. This could be a starting point of a deeper analysis in order to understand if it is a biomarker of Gilles de la Tourette syndrome.

The visual analysis of the differences between two bundle templates is more difficult than using meshes. Anyway it can also be useful if linked to the analysis of the internal structures and cortical surface. In fact, if one observes a difference between the templates of an internal structure (as with the left Caudate before), it's possible to check that particular area of the brain also looking at the bundle templates in order to understand exactly which are the parts involved in this change.



(a) Left Caudate

(b) Left Putamen

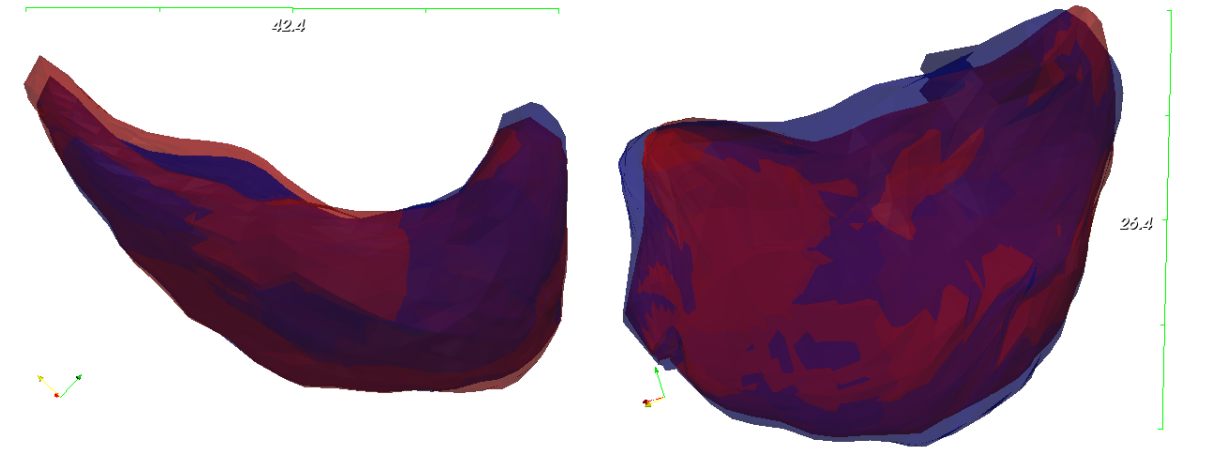
(c) Left Globus Pallidus bundle (left lateral view)

(d) Left Globus Pallidus bundle (right lateral view)

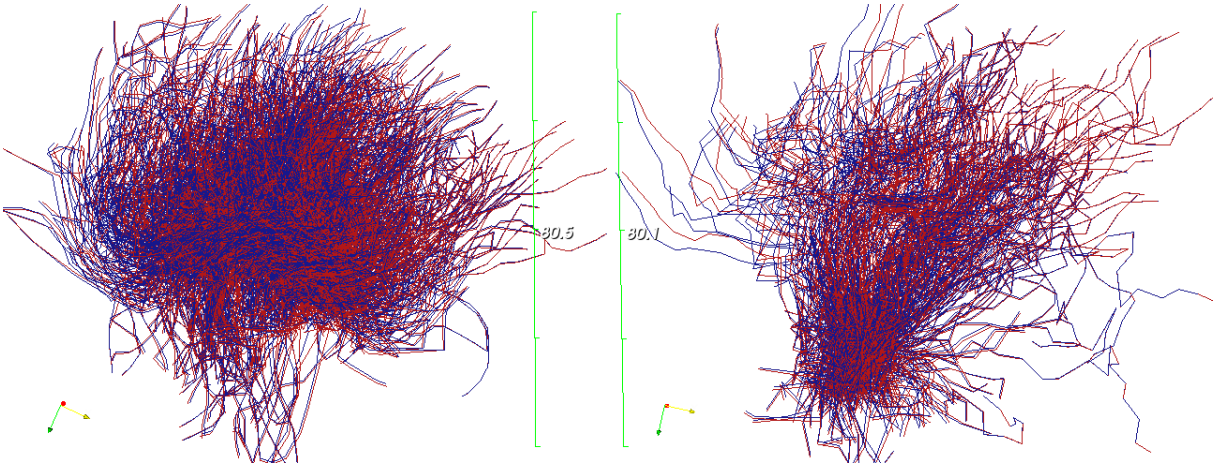
Figure 4.8: Comparison between the final templates of two different and independent Atlas Constructions using only controls in the first one (blue) and patients in the second one (red).

Analysis of the "mean deformations" As said before the parameters of each deformation are the sets of initial momenta $\alpha_i^s(0)$ that define the tangent-space of the diffeomorphism. If one computes the average values of the $\alpha_i^s(0)$ of each population (controls and patients), it's possible to find a kind of "mean deformation" which can characterize each population. In order to do that, we estimate the common template of both the populations with one Atlas Construction procedure. In this way, the template shows the common features among the two populations while the two "mean deformations" represent the traits of each population.

Looking at the results in Fig.4.9, it's possible to notice how they highlight the differences between the two populations in an easily understandable way. Also in this case we used only 8 subjects (4 controls and 4 patients) and therefore we can't draw any conclusion. Anyway it's easy to understand how this analysis can be really useful in a medical or neurological field since it permits to show directly the differences between the populations without giving as result a number or a function which can be difficult to "decipher" by a doctor.



(a) Left Caudate **(b)** Left Putamen



(c) Left Caudate bundle **(d)** Left Globus Pallidus bundle

Figure 4.9: Comparison between the "mean diffeomorphism" of controls (blue) and patients (red) starting from the same common template.

PCA To compute statistics on the deformations we take advantage of the tangent-space representation of the diffeomorphism which is given, as said before, by the sets of initial momenta. Each set determines one specific deformation of the template to an object and it can be seen as an "observation". Let X be a matrix in which each column is a different "observation", which means a set of initial momenta representing one deformation of the template to an object. Each column has a length of $3d$ where d is the number of control points (and therefore also momenta) and it can be seen as the concatenation of the 3 directions x,y,z of each momenta in a single vector. This matrix X has therefore a size of [$3d$ x N] where N is the number of subjects taken into account. More exactly, we compute twice the matrix X, once using only controls, and the second time with only patients. In fact, as before, we construct the Atlas using both the populations (finding therefore a common template) and after we look for the first mode of deformation considering independently controls and patients.

X is a matrix usually characterized by a high number of rows ($\alpha_i^s(0)$) and a relatively small number of columns (subjects). Thus, the computation of the eigenvectors and eigenvalues of the covariance matrix $\Sigma = \frac{1}{N}XX^T$ might be very difficult. A possible solution to this problem is to compute first $\tilde{\Sigma} = \frac{1}{N}X^TX$ which is smaller since its dimensions are no more [$3d$ x $3d$] but [N x N] and therefore also the computation of the eigenvalues and eigenvectors becomes easier and faster. Moreover it can be demonstrated that $\tilde{\Sigma}$ has the same eigenvalues different from zero as Σ. After that, calling $\tilde{\Lambda}$ this set of eigenvalues, $\tilde{\lambda}_i$ a single eigenvalue and $\tilde{V}$ the set of eigenvectors of $\tilde{\Sigma}$, it's possible to write:

$$\tilde{\Sigma}\tilde{V}_1 = \tilde{\lambda}_1\tilde{V}_1$$

$$\frac{1}{N}(X^TX)\tilde{V}_1 = \tilde{\lambda}_1\tilde{V}_1 \tag{4.29}$$

$$\frac{1}{N}XX^TX\tilde{V}_1 = \tilde{\lambda}_1X\tilde{V}_1 \tag{4.30}$$

$$\Sigma V_1 = \tilde{\lambda}_1V_1 \tag{4.31}$$

where we demonstrated that the eigenvectors of Σ are computable from the eigenvectors of $\tilde{\Sigma}$ and more exactly: $V = X\tilde{V}$.

Now, in order to be consistent with the rest of the project, we decided to compute the covariance matrix not with an Euclidean metric but with the RKHS metric used so far. This means that each element of $\tilde{\Sigma}$ is equal to $\left\langle(\alpha_i^s(0) - \overline{\alpha}), (\alpha_j^s(0) - \overline{\alpha})\right\rangle_W$ for every i and j between 1 and $3d$, where $\overline{\alpha}$ is the mean value of α in order to center the data. Now calling c_i a control point and K the Gaussian Kernel matrix it's possible to rewrite the previous equation as: $\tilde{\Sigma}^W_{i,j} = \sum_{i=1}^{3d}\sum_{j=1}^{3d}(\alpha_i^s(0) - \overline{\alpha})^TK(c_i, c_j)(\alpha_j^s(0) - \overline{\alpha})$.

In this way it's possible to compute the eigenvectors of $\tilde{\Sigma}^W$ and, multiplying them by X,

also the eigenvectors of Σ^W. The eigenvector relative to the biggest eigenvalue is exactly the first principal component which is a new direction in the α space. The aim of this process is to "shoot" in that direction the flow of the diffeomorphism in order to analyse the resulting deformation of the template. Before doing that we decided to normalize the eigenvector in order to know the "length" of the geodesic shooting. As usual we divided by the norm of the eigenvector multiplying by the standard deviation which is exactly the square root of the biggest eigenvalue:

$$\sqrt{\tilde{\lambda}_1}\frac{X\tilde{V}_1}{\|X\tilde{V}_1\|_W} \tag{4.32}$$

and since

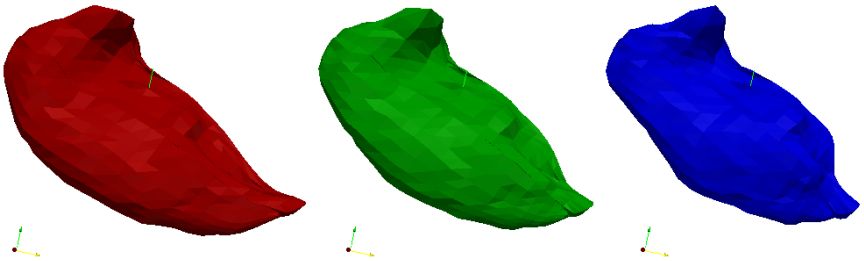
$$\|X\tilde{V}\|_W^2 = (X\tilde{V})^T K(X\tilde{V}) = \tilde{V}^T X^T K X\tilde{V} = N\tilde{\Lambda} \tag{4.33}$$

it can be written:

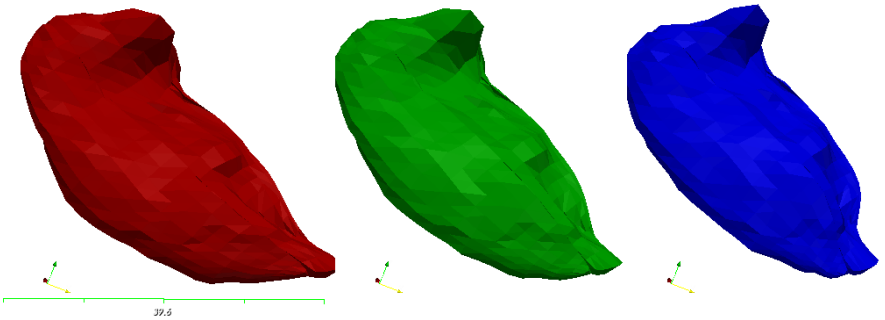
$$\sqrt{\tilde{\lambda}_1}\frac{X\tilde{V}_1}{\|X\tilde{V}_1\|_W} = \sqrt{\frac{1}{N}}X\tilde{V}_1 \tag{4.34}$$

Using the first eigenvector normalised it's possible to find the direction which explains most of the diffeomorphic variability (first mode of deformation) and multiplying it by the standard deviation we obtain a deformation of the template within the 90% range of the shape variation within the population.

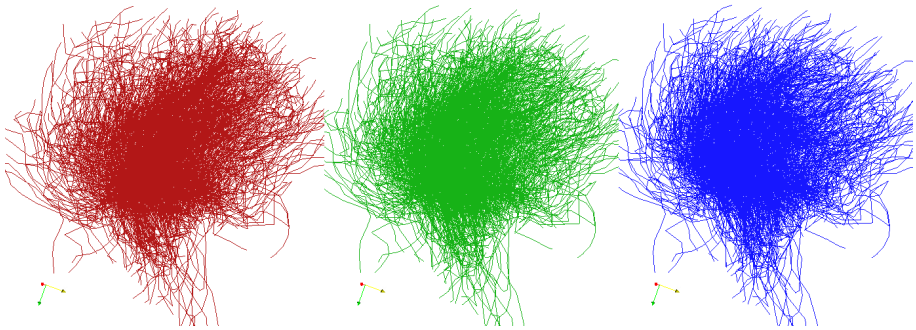
In Fig.4.10 examples of first mode of deformation at $\pm\sigma$ are shown near the template. We built a common template using both the populations and after we compute the first principal component considering only the set of initial momenta of controls or patients. Deforming the template with the resulting set of initial momenta, we can check the main variability of the common template in both the populations. Looking at the first two figures about the Left Caudate, we notice that there is a stretching/shortening of the small horn in both the populations while there is a reduction/expansion of the part opposite to the horn only in the Controls. The other two figures regard the left Caudate bundle. Here the main variability seems to be a torque in the upper/middle part of the bundle in both populations. But the population of patients is also characterised by a movement of fibers from left to right and viceversa in the middle area.



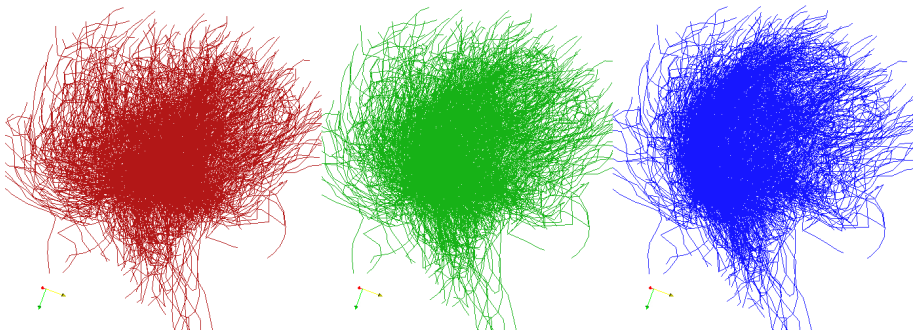
(a) Left Caudate - CONTROLS - First mode deformation at -σ (red), template (green) and first mode deformation at +σ



(b) Left Caudate - PATIENTS - First mode deformation at -σ (red), template (green) and first mode deformation at +σ



(c) Left Caudate bundle - PATIENTS - First mode deformation at -σ (red), template (green) and first mode deformation at +σ



(d) Left Caudate bundle - PATIENTS - First mode deformation at -σ (red), template (green) and first mode deformation at +σ

Figure 4.10: First mode of deformation at $\pm\sigma$ using only momenta of controls or patients.

4.8 Conclusion about Atlas Construction

In this section it was shown an atlas construction method to build a common template and relative deformations from a population of mesh and/or curves. The template shows the common features of the population while the parametrizable deformations of the template into each object permit to characterize the variability of the population. The deformation framework used was based on one diffeomorphism of the space: a smooth one-to-one deformation of the 3D space with a smooth inverse. It means that, using different kind of objects (mesh or curves), we deformed the 3D space in order that each deformed template (one for each kind of object) best matches all the objects present in the population. Thus, given for example M different objects of N subjects, we have in total MN objects and N templates. The goal of the Atlas Construction method is to estimate the N templates and their relative MN deformations using a diffeomorphism of the whole underlying space.

We decided to use for each deformation the same set of the control points but different initial momenta ($\alpha(0)$) which represent its parametrization. Since they build a vector space it's also possible to compute statistics on it. We showed an example computing the first principal component at $\pm\sigma$ considering only patients or controls and using the resulting sets of initial momenta to deform a common template. In this way it's possible to check which is the main variability of both the populations.

In order to compare the two populations of controls and patients we used also other two techniques: we compare the templates resulting from two different and independent Atlas constructions using only controls or patients and we compare the "mean deformations" of a common template of both the populations. In both cases it's easy to check the differences just with a visual inspection of the results. This can be really useful in medical and neurological field since it permits to the neuroscientists to see directly the results and the differences between the two populations. Therefore it can be seen as a good starting point for a deeper and more rigorous research in the area that show the main differences to drive the search for hypothesis on the anatomical underpinnings of neurological pathologies.

We have demonstrated that this technique can be used not only for mesh (already shown in [DPK+12]) but also for curves, in this case fiber bundles. This was exactly one of the goal of this Master's Thesis. The other main aim was to check the hypothesis of Worbe et al. [WGH+10] linked to Gilles de la Tourette syndrome and explained in Chapter 2. This is a project much bigger than a Master's Thesis and it will be indeed a future PhD project. In fact, such an analysis requires more tools, especially statistical, for analysing the results and above all it needs more time. It was pretty slow the analysis of the data (i.e. 200 hours for one atlas construction with 1 bundle and 10 subjects) and this is another challenging issue to solve in a future development.

Nevertheless we managed to demonstrate the usefulness of this method also applied to

the bundles and we showed a lot of possible ways to improve it. Future studies will focus mainly on an automatic way for choosing the parameters and on the selection of suitable statistical tests.

Chapter 5

Conclusions

The atlas construction method shown for the first time in the paper of Durrleman et al. [DPK+12] and tested only on surfaces was successfully implemented also for fiber bundles modelled as 3D *curves*. The data set of this project is composed by cortical surfaces and sub cortical structures modelled as 3D meshes and by fiber bundles of 74 subjects divided in 27 controls and 47 patients with Gilles de la Tourette syndrome.

One of the main goals of this project was to lay the groundwork for a deeper study of the hypothesis presented in Worbe et al. [WGH+10] about Gilles de la Tourette syndrome. We checked the differences between a population of controls and one of patients in the cortico-striato-thalamo-cortico circuits composed by white matter tracts (fiber bundles) and also in the linked cortical/sub cortical areas. We could not draw any neuroscientific conclusion from our results due to the limited number of subjects involved in the experiments. But we showed how to check and interpret the differences between the two populations using the atlas construction method presented.

Moreover we showed also how to build an atlas using *together* 3D curves and 3D meshes in a way which preserved their topology. In order to study the hypothesis linked to GTS it is indeed necessary to have a bundle template which keeps the connectivity between points. During the atlas construction, a template and its deformations to each object of the population are estimated. The analysis of the deformations can give an insight into the variability of the population. If a bundle template is not composed by fibers (i.e. without connectivity between points), it's much more difficult to keep track of the motion of its starting and ending points and therefore also to find alterations in the neural connectivity. This makes less feasible the analysis of the neural circuits and therefore also the test of the hypothesis of Worbe et al.

We applied this methodology using both populations (controls and patients) or separately. First we constructed two atlases comparing the templates since they show the common features of each population. Looking at the results, if the number of subjects had been bigger, we could have drawn the conclusion that the sub cortical objects seem to be bigger in the population of patients while it's really difficult to see any clear difference between the templates of the bundles. We are still investigating a better way for displaying them in order to simplify the comprehension of their differences.

After that we also built one single atlas taking into consideration both the populations and we compare the deformations of the common template to the objects of one population with respect to the deformations to the objects of the other population. In this way we checked the variability of the common template in both populations. In order to do that, we compared first the "mean deformations" to each population and after also the deformations which explain most of the variability of the common template in both populations (i.e. the first principal component). Also in this case the results weren't meaningful since the number of subjects wasn't adequate, but we noticed some differences between the populations that could have driven the search for markers of GTS. It's also important to highlight the fact that looking at the first principal component at $\pm\sigma$ helped a lot the analysis of the fiber bundles. It was the only technique that showed pretty clearly some differences between the two populations.

As said before, this Master's Thesis represents a first step in a longer and deeper research of GTS markers which will continue with a PhD. Throughout this project we showed the merits but also the limitations of the Atlas construction method proposed. Future developments will focus mainly on an automatic choice of the parameters, on the selection of suitable statistical tests and on the addition of more subjects. This will make our results more precise and above all more meaningful and hopefully statistically significant. Moreover we will include also the cortical surfaces in this study, which is fundamental in order to check the change in connectivity between different gray matter structures.

Appendix A

AppendixA

A.1 Vector Spaces

Considering **V** a set (not empty) and **K** a field (an algebraic structure with notions of addition, subtraction, multiplication and division satisfying certain axioms) with the following two operations:

1. addition in **V**, which means a correspondence between each couple of elements **(x,y)** belonging to **V** and another element of **V** called sum of **x** and **y** and represented by **x**+**y**.
2. multiplication of the elements in K by the elements in **V**, which means a correspondence between each couple of elements (a,**x**) with a $\in$ **K** and **y** $\in$ **V** and another element of **V** called product and represented by a**x**.

Definition A.1 one can call **V** a vector space over the field **K** if and only if the operations respect the following properties:

- Commutative property : **x**+**y**=**y**+**x**
- Associative property : (**x**+**y**)+**z**=**x**+(**y**+**z**) and (ab)**x** = a(b**x**)
- Existence of Neutral element for addition : it's an element in **V** called **0** in order that **0**+**x**=**x** $\forall$ **x**.

- Existence of Neutral element for multiplication : it's an element in **K** called 1 in order that 1**x**=**x** $\forall$ **x**.
- Existence of the opposite : $\forall$ **x** it exists an element **y** such that **x** + **y** = **0**.
- Distributive property of the multiplication with respect to the addition between vectors in **V** and scalars in **K** : a(**x**+**y**)=a**x**+a**y** and (a+b)**x**=a**x**+b**x**.

It's important to notice that all the elements of **V** are vectors and all the elements of **K** are scalars. **K** is usually chosen as $\mathbb{R}$ or $\mathbb{C}$.

Being **V** a v.s. (vector space) over a field **K**; if **x** and **y** are two vectors and a,b two scalars, the vector a**x**+b**y** is called linear combination of **x** and **y** with the coefficients a and b.

Definition A.2 If **V** is a vector space over **K** with the previous defined operations, and **W** is a subset of **V**; it is possible to define **W** as a subspace of **V** if:

$$(\text{a,b} \in \mathbf{K} \wedge \mathbf{x},\mathbf{y} \in \mathbf{W}) \Longrightarrow \text{a}\mathbf{x} + \text{b}\mathbf{y} \in \mathbf{W}$$

Practically also **W** is a vector space with the same operations and hypothesis as before and it can be called v.s. if and only if it contains all the linear combinations of vectors belonging to it. There are two kinds of subspaces of **V**: the proper subspace and the improper subspace. The only two examples of the latter are **V** itself and the subspace composed only by the null vector **0**, while all the other possible subspaces are called proper.

Definition A.3 The vectors $\vec{x_1}$,$\vec{x_2}$,...,$\vec{x_n}$ of the v.s. **V**are called *linearly independent* if and only if it follows from:

$$\mathbf{0} = a_1\vec{x_1} + a_2\vec{x_2} + ...a_n\vec{x_n} \tag{A.1}$$

that $a_1 = a_2 = ... = a_n = 0$. Otherwise the vectors are called *linearly dependent*.

From the previous definition it follows that if the vectors are linearly independent all of them are different from the null vector and if they are linearly dependent one can express the vectors with the coefficients different from zero as a linear combination of the other vectors. In $\mathbb{R}^2$ two vectors are linearly independent if they are not null and not proportional, which means that they don't belong to the same line passing through the origin. Instead, three vector of $\mathbb{R}^3$ are linearly independent if and only if they are different from the null vector and they are not coplanar.

Definition A.4 A set of vectors $\vec{x_1}$,$\vec{x_2}$,...,$\vec{x_n}$ is called a *basis* of **W**if :

- **W**is generated by this set of vectors, which means that is equal to all the linear combinations of the vectors or that this set of vectors spans the whole space of **W**. Practically any vector **x**can be expressed as a finite sum of the basis vectors (if the basis is finite).
- and if the set of vector is minimal which is made formal by requiring the set of vectors to be linearly independent.

Linear independence ensures that the representation of any vector in terms of basis vectors is unique and the existence of a linear combination of the basis vectors equal to the vector chosen is guaranteed by the requirement that the basis spans **W**.

For all the sets of vectors that span **W** is possible to find a basis, discarding the null vectors and all the vectors that can be written as linear combination of the vectors already accepted in the basis. This means that it's possible to have more than one basis for each v.s. **W** but it can be demonstrated that all the basis of **W** must have the same number of vectors. This number is called *dimension* of **W**.

It can also be demonstrated the following important proposition:

Proposition A.5 *Being* $\vec{x_1}$,$\vec{x_2}$,...,$\vec{x_n}$ *n vectors of a v.s.* **V** *of dimension n, the two following conditions are equivalent:*

1. $\vec{x_1}$,$\vec{x_2}$,...,$\vec{x_n}$ *span* **V**
2. $\vec{x_1}$,$\vec{x_2}$,...,$\vec{x_n}$ *are linearly independent*

Linear maps or Linear Transformations

Definition A.6 Let **V** and **W** be two vectors spaces over the same field **K**. A function $f : \mathbf{V} \to \mathbf{W}$ is said to be a *linear map* if for any vector $\vec{x_1}$,$\vec{x_2}$,...,$\vec{x_n}$ $\in$**V** and any scalar a_1,a_2,...,$a_n \in \mathbf{K}$, the following equality holds:

$$\mathrm{f}(a_1\vec{x_1}+...+a_n\vec{x_n}) = a_1\mathrm{f}(\vec{x_1})+...+a_n\mathrm{f}(\vec{x_n})$$

This is equivalent to requiring that: for any two vectors $\vec{x}$ and $\vec{y} \in \mathbf{V}$ and any scalar $a \in \mathbf{K}$, the following two conditions are satisfied:

1. $f(\vec{x}+\vec{y})=f(\vec{x})+f(\vec{y})$

2. $f(a\vec{x}) = af(\vec{x})$

So practically a linear map is a function that preserve the vector space structure, which means the operations of **V** and **W** based on the same field **K**. From a practical point of view, if **V** and **W** are finite-dimensional with their respective bases also finite-dimensional, a linear map f from **V** to **W** can be represented as a matrix. In fact:

- Any vector $\vec{v} \in$**V** can be represented as a linear combination of each basis of **V** $(a_1\vec{x_1}+a_2\vec{x_2}+...+a_n\vec{x_n})$, which means that v is uniquely determined by the coefficients $a_1, a_2, ..., a_n$.

- Using A.6, $f(a_1\vec{x_1}+a_2\vec{x_2}+...+a_n\vec{x_n})$ can be rewritten as: $a_1f(\vec{x_1})+...+a_nf(\vec{x_n})$

- And all the functions present in the second part of the previous equation can be rewritten as linear combination of any basis $(\vec{y_1},...,\vec{y_m})$ of **W** : $f(\vec{x_j})=c_1j\vec{y_1}+c_2j\vec{y_2}+...+c_mj\vec{y_m}$

- So one can write: $f(\vec{v})= a_1(c_{11}\vec{y_1}+...+c_{m1}\vec{y_1})+...+a_n(c_{1n}\vec{y_1}+...+c_{mn}\vec{y_1})$

Thus the function f of any vector present in any basis of **V** is entirely determined by the coefficients a_j and c_{ij}. The last equation can also be written in matrix notation as $f(\vec{v})=$**CA** where **C** is the matrix m-*by*-n containing all the c_{ij}, while **A** is a column vector containing all the a_j. The result is a column vector containing the coefficients of all the basis vectors $\vec{y_i}$. Of course a linear map can be represented in many ways, since the matrix **CA** depends on the choice of the basis.

Using a more general term from a mathematical point of view, one could say that a linear transformation $f : \mathbf{V} \to \mathbf{W}$ is an endomorphism of **V** , which means a mathematical structure that preserves the mapping from a mathematical object to itself. This means that it's a morphism (or homomorphism) where the mapping is between two identical algebraic structures and not between two generic ones.

In the case that the morphism is also bijective, f can be called isomorphism from **V** to **W**. It can be said also that **V** and **W** are isomorphic if it exists an isomorphism which transforms **V** into **W** (and since it is bijective also **W** in **V**).

An endomorphism of **V** which is also an isomorphism is called automorphism of **V**.

Normed vectored space

Definition A.7 Let $\mathbf{V}$ be a v.s. real or complex (which means that $\mathbf{K}\subseteq\mathbf{R}^n$ or $\mathbf{K}=\mathbf{C}$), it is called *norm* on $\mathbf{V}$the function $\|\| : \mathbf{V} \to \mathbf{R}$ with the following properties:

1. $\mathbf{x} \neq \mathbf{0} \Longrightarrow \|\mathbf{x}\| > 0$
2. $\|a\mathbf{x}\| = |a|\ \|\mathbf{x}\|$, $\forall a \in \mathbf{K}$, $\mathbf{x} \in \mathbf{V}$
3. $\|\mathbf{x}+\ \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$, $\forall \mathbf{x}, \mathbf{y} \in \mathbf{V}$

The pair ($\mathbf{V}$,$\|\cdot\|$) is called *normed vector space*. The first two properties imply that $\|\mathbf{x}\|$ = 0 if and only if $\mathbf{x} = \mathbf{0}$. Instead a useful variation of the third property (called "triangle inequality") is $\|\mathbf{x}-\ \mathbf{y}\| \geq \|\mathbf{x}\|$ - $\|\mathbf{y}\|$.

The most used and general form of norm is:

$$\|x\|_p = \left(\sum_{k=1}^{n} |x_k|^p\right)^{1/p}$$

where, when $p = 2$, it becomes the most commonly used norm on $\mathbf{R}^n$ called "Euclidean norm" or when $p = 1$ it is called the Manhattan norm or L_1 norm. Instead another important norm is $\|\mathrm{x}\|_\infty$ which is called Maximum norm or also supremum norm or infinity norm and it is defined as $\|\mathrm{x}\|_\infty \doteq \max(|x_1|, ..., |x_n|)$.

A *normed vector space* is automatically also a *metric space* where the *metric* (or distance function) is a function which defines a distance between elements of the set, in this case vectors, and it is defined as:

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| \tag{A.2}$$

In order to check if the norm can be a distance function it is useful to recall the definition of distance:

Definition A.8 Let $\mathbf{Y}$ be a set (not empty) and d a function $d : \mathbf{Y}\mathrm{x}\mathbf{Y} \Longrightarrow \mathbf{R}$, a *metric* on $\mathbf{Y}$ is a function that satisfies the following conditions:

1. $\mathrm{d}(\mathbf{x},\mathbf{y}) \geq \mathbf{0} \quad \forall\ \mathbf{x},\mathbf{y} \in \mathbf{Y}$

2. $\mathrm{d}(\mathbf{x},\mathbf{y}) = \mathbf{0} \iff \mathbf{x}=\mathbf{y} \quad \forall\, \mathbf{x},\mathbf{y}\in \mathbf{Y}$
3. $\mathrm{d}(\mathbf{x},\mathbf{y}) = \mathrm{d}(\mathbf{y},\mathbf{x}) \quad \forall\, \mathbf{x},\mathbf{y}\in \mathbf{Y}$
4. $\mathrm{d}(\mathbf{x},\mathbf{y}) \leq \mathrm{d}(\mathbf{x},\mathbf{z}) + \mathrm{d}(\mathbf{z},\mathbf{y}) \quad \forall\, \mathbf{x},\mathbf{y},\mathbf{z}\in \mathbf{Y}$

From this definition it is easy to demonstrate that Eq. A.2 is correct and that it can be applied to all the *Normed vector spaces*. It must be also clear that different norms induce different distances in the v.s.

Another important concept in the metric space is the one of "*Balls*".

Definition A.9 Let $(\mathbf{V},\|\cdot\|)$ be a n.v.s. (normed vector space); the *open ball* of radius $r>0$ centered at $\mathbf{p} \in \mathbf{V}$and denoted as $\mathbf{B}_r(\mathbf{p})$, is defined by:

$$\mathbf{B}_r(\mathbf{p}) := \mathbf{x} \in \mathbf{V} : \|\mathbf{x}-\mathbf{p}\| < r \tag{A.3}$$

Instead the *closed ball*, denoted as $\mathbf{B}_r[\mathbf{p}]$, is defined by:

$$\mathbf{B}_r[\mathbf{p}] := \mathbf{x} \in \mathbf{V} : \|\mathbf{x}-\mathbf{p}\| \leq r \tag{A.4}$$

Consequently, since any *normed vector space* is also a metric space, every ball $\mathbf{B}_r(\mathbf{p})$ can be considered as a copy of the unit Ball $\mathbf{B}_1(\mathbf{0})$, scaled by r and translated by $\mathbf{p}$. For example, in Fig. A.1 are showed three kind of unit balls for the norms $\|\cdot\|_p$ with $p = 1,2$ and ∞.

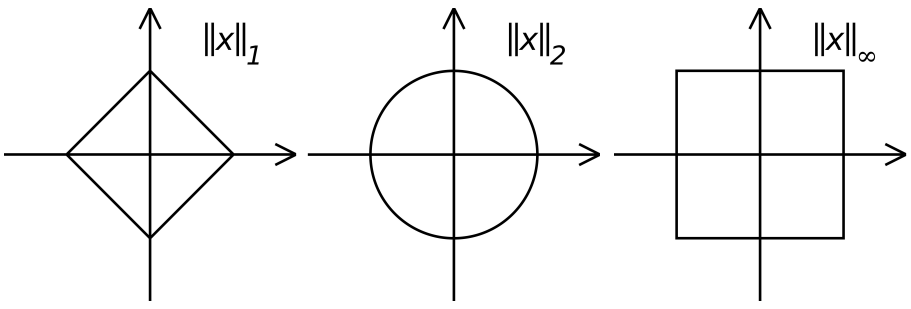


Figure A.1: Unit Balls centred in $\mathbf{0}$ with unit ray for three different norms $\|\cdot\|_p$ with p= 1,2 and ∞.

Another important concept linked with the definition of "Ball" is the one of equivalence between norms.

Definition A.10 Being $\|\cdot\|_1$ and $\|\cdot\|_2$ two norms over the same n.v.s. $\mathbf{V}$, they are said to be *equivalent* if every ball around a point using one norm contains a ball of the other norm, which means that there are two constants c_1 and c_2 for which:

$$\|\mathbf{x}\|_1 \leq c_1\|\mathbf{x}\|_2 \; and \; \|\mathbf{x}\|_2 \leq c_2\|\mathbf{x}\|_1 \quad \forall \mathbf{x} \in \mathbf{V} \tag{A.5}$$

Using this definition it can be demonstrated that in a n.v.s. of finite dimension (like **R**or **C**) all the norms are equivalent, which means for example that:
$\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1 \leq \mathrm{c}\|\mathbf{x}\|_\infty$.

After the definition of *Open Ball* it is possible to extend the usual topological notions to all the *normed vector spaces*.

In fact, as said before, all the *normed vector spaces* are also *metric spaces* which are *topological spaces* in a natural manner since the Open Balls defined before form the *basis* **B**for a topology **T**on the metric space, making it a topological space **X**.

A *basis* **B** for a topological space **X** with topology **T** is a collection of open sets in **T** such that every open set in **T** can be written as a union of elements of **B**. Strictly speaking one could say that the *basis* **B** generates the topology.

In order to better understand this part it is useful to recall the definition of *Topology* and *Topological Space*.

Definition A.11 Let **X** be a set and let **T** be a family of subsets of **X**. Then **T** is called a *topology* on **X** if:

1. $\emptyset$,$\mathbf{X} \subset \mathbf{T}$: both the empty space and **X** are elements of **T**.
2. $(\mathbf{T_1}, \mathbf{T_2}, ...\mathbf{T_n}) \subseteq \mathbf{T}$ implies $(\mathbf{T_1} \cup \mathbf{T_2} \cup ... \cup \mathbf{T_n}) \subseteq \mathbf{T}$: Any union of elements of **T** is an element of **T**.
3. $(\mathbf{T_1}, \mathbf{T_2}, ...\mathbf{T_n}) \subseteq \mathbf{T}$ implies $(\mathbf{T_1} \cap \mathbf{T_2} \cap ... \cap \mathbf{T_n}) \subseteq \mathbf{T}$: Any intersection of a finite number of elements of **T**is an element of **T**.

The pair (**X**,**T**) is called a topological space and all the members of **T** are open sets in **X**. An open set **Y** is a set consisting only of interior points which means that **Y** contains a Ball of a certain ray around all the interior points. From this definition it's possible to see the link between Open Balls and Open Sets and thus why a metric space is automatically a topological space.

Definition A.12 A normed vector space $\mathbf{V}$ is called *complete* if every Cauchy sequence defined in $\mathbf{V}$ converges to an element of $\mathbf{V}$.

Definition A.13 A sequence (x_n) of elements of a normed vector space $\mathbf{V}$ is called *Cauchy sequence* if, for every $\epsilon>0$, it exists an index p (which depends on ϵ) for which:

$$\forall n, m > p \Longrightarrow \|x_n - x_m\| < \epsilon$$

Thus a normed vector space $\mathbf{V}$ is called *complete* if for every Cauchy sequence (x_n) it exists a point $x \in \mathbf{V}$ where the sequence converges. Or, in other words, the elements of the sequence become arbitrarily close to each other (depending on ϵ) as the sequence progresses. All the complete normed vector spaces are also called *Banach spaces*. And it can be demonstrated that:

Proposition A.14 *Every normed vector space over* $\mathbf{R}$ *or* $\mathbf{C}$ *characterized by a finite dimension is complete, which means a Banach space.*

Inner product space

In mathematics an *inner product space* (or *pre-Hilbert space*) is a vector space $\mathbf{V}$ with an additional structure called inner product.

Definition A.15 The inner product related to the *inner product space* $\mathbf{V}$ is a function $\langle\cdot\rangle : \mathbf{V} \times \mathbf{V} \rightarrow \mathbf{K}$ (where $\mathbf{K}$ can be $\mathbf{R}$ or $\mathbf{C}$) that satisfies the following properties: $\forall\, \mathbf{x},\mathbf{y},\mathbf{z} \in \mathbf{V}$ and $\alpha \in \mathbf{K}$:

- $\langle \mathbf{x},\mathbf{y} \rangle = \overline{< \mathbf{y}, \mathbf{x} >}$
- $\langle \alpha\mathbf{x},\mathbf{y} \rangle = \alpha \langle \mathbf{x},\mathbf{y} \rangle$
- $\langle \mathbf{x},\mathbf{y} + \mathbf{z} \rangle = \langle \mathbf{x},\mathbf{y} \rangle + \langle \mathbf{x},\mathbf{z} \rangle$
- $\mathbf{x} \neq \mathbf{0} \Longrightarrow \langle \mathbf{x},\mathbf{x} \rangle > \mathbf{0}$

From these axioms it follows that $\langle \mathbf{x},\mathbf{0} \rangle = 0$ and from the first property it follows that $\langle \mathbf{x},\mathbf{x} \rangle \in \mathbf{R}$also if $\mathbf{K} = \mathbf{C}$. Moreover, if $\mathbf{K} = \mathbf{R}$it is trivial to see from the first two axioms that the inner product is symmetric.

Definition A.16 Two vectors $\mathbf{x},\mathbf{y}$ of an *inner product space* $\mathbf{V}$ are called orthogonal if $\langle \mathbf{x},\mathbf{y} \rangle = 0$.

In every *inner product space* it is defined in a natural way a norm generated or induced by the inner product itself. This is : $\|\mathbf{x}\| := \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$. The first two properties of the definition of norm (Def. A.7) are straightforward but, in order to demonstrate the third condition (triangular inequality), it is necessary to introduce the Theorem of the Cauchy-Schwarz inequality:

Theorem A.17 *If* $\mathbf{H}$ *is a inner product space and* $\mathbf{x}$,$\mathbf{y} \in \mathbf{H}$*, it is true the following condition:*

$$|\langle \mathbf{x}, \mathbf{y} \rangle|^2 \leq \langle \mathbf{x}, \mathbf{x} \rangle \cdot \langle \mathbf{y}, \mathbf{y} \rangle \tag{A.6}$$

which is equal to:

$$|\langle \mathbf{x}, \mathbf{y} \rangle|^2 \leq \|\mathbf{x}\|^2 \cdot \|\mathbf{y}\|^2 \tag{A.7}$$

Moreover, the equal sign is valid if and only if $\mathbf{x}$ and $\mathbf{y}$ are linearly dependent (which means if they are parallel or one of them is equal to $\mathbf{0}$).

Definition A.18 A *Hilbert Space* $\mathbf{H}$is a real or complex inner product space which is also complete with respect to the norm induced by the inner product.

Since an inner product space $\mathbf{V}$can also be considered a normed vector space (the inner product always induces a norm), if it's of finite dimension, using Preposition A.14, it can be considered also complete and thus a *Hilbert Space*.

Probably the most famous example of Hilbert space is the l^2 space. This is the set of all infinite sequences (x_i) of numbers (real or complex) such that the series $\sum_{i=1}^{\infty} |x_i|^2 < \infty$. Furthermore this set has the structure of a vector space by applying addition and scalar multiplication coordinate by coordinate:

$$\alpha(x_i) := (\alpha x_1, \alpha x_2, ..., \alpha x_n, ..) \quad \textit{and} \quad (x_i) + (y_i) := (x_1 + y_1, x_2 + y_2, ..., x_n + y_n, ...)$$

with α,$\beta \in \mathbf{R}$or $\mathbf{C}$. Moreover, in the the l^2 space, it can also be defined the inner product:

$$\langle (x_i), (y_i) \rangle = \sum_{i=1}^{\infty} x_i \overline{y_i}$$

And thus also the induced norm:

$$\|(x_i)\| = \sqrt{\sum_{i=1}^{\infty} |x_i|^2}$$

Eventually one can also demonstrate that l_2 is complete with respect to the norm induced by the inner product and thus it's an Hilbert Space.

A.2 Formal mathematical definition of Currents

The concept of *current* was introduced by Georges de Rham in the 20th century [dR73] and it was mainly based on the work of another great mathematician, L. Schwartz, who built up the whole theory of distributions [Sch66]. Distributions are a class of linear functionals[1] that map a set of *test functions*[2], infinitely differentiable (smooth) and with a compact support, onto the set of real numbers. Thus a distribution is a linear mapping from the set of test functions to $\mathbf{R}$. Instead a current is a linear functional that maps a space of smooth m-forms with compact support onto $\mathbf{R}$ and the functional is continuous in the distribution sense.

A m-form w on $\mathbf{R}^d$ is a linear map from the vector space spanned by m vectors $\in \mathbf{R}^d$ via the wedge product $\wedge$ to $\mathbf{R}$. The wedge product is a generalization of the cross-product which extends the usual measure of area and volume in 3D [Dur10] and it can be defined by the following properties:

$$\begin{cases} (\lambda u + v) \wedge w = \lambda(u \wedge w) + v \wedge w \\ w \wedge (u + v) = w \wedge u + w \wedge v \\ u \wedge u = 0 \\ u \wedge v = -v \wedge u \\ u \wedge v \wedge w = (u \wedge v) \wedge w \end{cases}$$

where $u, v, w \in \mathbf{R}^d$ and $\lambda \in \mathbf{R}$. Therefore the wedge product is a bilinear anticommutative operation which vanishes if two vectors are equal and it is provided with an associativity law. The result of a wedge product between two vectors is no more a vector but a bivector, between three vectors it is called trivector and between m vectors it is a m-multivector. As said before, the wedge product $u \wedge v$ is similar to the cross product u x v but it doesn't require the vector space to have a metric while the cross product necessarily assumes a metric since it generates a vector orthogonal to u and v [Bro09]. Moreover the exterior (wedge) product is not only confined to products of two vectors in 3D but is valid in any dimension and also for more than 2 vectors. For example, $u \wedge v \wedge w$ is a trivector whose magnitude is equal to the triple product between the three vectors which means the signed volume of the parallelepiped created by the three vectors.

This leads to the definition of the vector space of the *m-forms* spanned by m vectors via the wedge product $(u_1 \wedge u_2 ... \wedge u_m)$, usually denoted as $\Lambda^m \mathbf{R}^d$, and called the *m-th* exterior power of $\mathbf{R}^d$ whose vectors are called m-multivectors. The space of m-forms is therefore the dual space of $\Lambda^m \mathbf{R}^d$ since it is the set of all linear maps from the vector space $\Lambda^m \mathbf{R}^d$ to

[1] linear map from a vector space to its field of scalars

[2] Test functions are functions compactly supported and continuously differentiable of any order $D = C^\infty(\mathbf{R})$. A distribution is a continuous linear functional on every test function of D and the action of any functional f on a test function ϕ is given by $\langle f, \phi \rangle = \int f(x)\phi(x)dx$.

its field $\mathbf{R}$. It can be shown that each multivector that spans the vector space of a m-form can be decomposed into a sum of the canonical basis of $\mathbf{R}^d$ and thanks to the linearity and alternating properties of the wedge product it can also be demonstrated that $\Lambda^m\mathbf{R}^d$ is spanned by the canonical basis of $\mathbf{R}^d$. When m is equal to 1, $\Lambda^m\mathbf{R}^d$ is the vector space $\mathbf{R}^d$ which can be provided of the standard Euclidean inner-product and norm, becoming an *inner product space*, moreover since it is of finite dimension (at least in this project) it is also *complete* and therefore it can be considered a *Hilbert Space*. Thus, thanks to the Riesz representation theorem [Rie09], every element of the 1-form can be represented by the inner product with a fixed vector $\overline{w}$ such as: $w(u) = \overline{w}^T u$ where u is a vector in $\mathbf{R}^d$ and it belongs to $\Lambda^1\mathbf{R}^d = \mathbf{R}^d$ and $w(u) \in \mathbf{R}$.

Now it's useful to introduce the concept of differential m-forms on $\mathbf{R}^d$. They map every point $x \in \mathbf{R}^d$ to $w(x)$ a m-form in $(\Lambda^m\mathbf{R}^d)^*$ which is the space of m-forms on $\mathbf{R}^d$. The space of differential m-forms which are continuous and tend to zero at infinity is called $C^0(\mathbf{R}^d, (\Lambda^m\mathbf{R}^d)^*)$ and it is also provided with a supremum norm $\|\mathrm{w}\|_\infty$:

$$\|w\|_\infty = \sup_{x\in R^d} \sup_{|u_1\wedge\ldots\wedge u_m|\leq 1} |w(x)(u_1 \wedge \ldots \wedge u_m)| \tag{A.8}$$

Using this last concept, it can be provided another definition of *current*: the dual space of the space of differential m-forms $C^0(\mathbf{R}^d, (\Lambda^m\mathbf{R}^d)^*)$. With this last definition it is enhanced the link with the Schwartz distributions, and above all the continuity of the functional and the compactness of the support. In fact currents behave like distributions but on a space of *differential m-forms* instead of *test functions*. And, when m is equal to zero, the differential 0-form is a scalar field and thus a 0-current is simply a Schwartz distribution.

Instead when m is equal to one, a differential 1-form may be represented by a vector field $\overline{w}$ such that for all vectors $u \in \mathbf{R}^d$ and for all points $x \in \mathbf{R}^d$ it can be written: $w(x)(u) = \overline{w}(x)^T u \in \mathbf{R}$. Practically, for each point x there is a vector attached u which is linked to a scalar value due to the m-form. Moreover, thanks to the Riesz representation theorem, this linear form can be represented by an inner product with a fixed vector which is in this case the vector field $\overline{w}(x)$ for every point x.

This result is a key point for this project, and it can be demonstrated that when m=1 the differential 1-form is a vector field whatever is the value of d. In this project d will take only the value of 3 which means that all the vector fields will belong to a 3D space. Reminding the definition of currents as a *continuous linear map from a space of differential m-forms to* $\mathbf{R}$, it's useful to give some practical examples.

Let **L** be a piecewise differentiable curve, it can be seen also as a 1-current via the equation:

$$L(w) = \int_L w(x)^T \tau(x) d\lambda(x) \tag{A.9}$$

for every vector field w, where $\tau(x)$ is the unit tangent vector of the curve **L** at point x. In this last equation $d\lambda(x)$ denotes the Lebesgue measure on the curves so that $\int_L d\lambda(x)$ equals the total length of the curve. w is a vector field that plays the role of the 1-form.

As explained in [Dur10] also a differential 2-form (m=2) in dimension 3 ($d = 3$) is of dimension 3 and thus it may be represented by a vector field $\overline{w}(x)$. Thus each 2-form w is associated to a 3D vector field such that $w(u, v) = det(u, v,)\overline{w} = \overline{w}^T(u \text{ x } v)$. From a practical point of view, a 2-form in dimension 3 is exactly a surface in 3D which is the second object used throughout this project together with the curves in 3D (m=1, d=3).

Let **S** be a set of piecewise differentiable surfaces, it can be seen as a 2-current via the following equation:

$$S(w) = \int_S w(x)^T (u(x) \times v(x)) d\lambda(x) = \int_S w(x)^T \eta(x) d\lambda(x) \tag{A.10}$$

for every vector field w, where $\eta(x) = (u(x) \times v(x))$ is the unit normal vector of the surfaces S at point x, $(u(x) \times v(x))$ is the orthonormal basis of the tangent plane of the surface S at point x.

Bibliography

[AAT07] Stéphanie Allassonnière, Yali Amit, and Alain Trouvè. Towards a coherent statistical framework for dense deformable template estimation. *Journal of the Royal Statistical Society Series B*, 69(1):3–29, 2007.

[ADS86] Garrett E. Alexander, Mahlon R. DeLong, and Peter L. Strick. Parallel organization of functionally segregated circuits linking basal ganglia and cortex. *Ann. Rev. Neurosci.*, 9:357–381, 1986.

[Ass00] American Psychiatric Association. *Diagnostic and statistical manual of mental disorders*. DSM-IV-TR, 4th edition, 2000.

[BMTY05] Mirza Faisal Beg, Michael I. Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 61(2):139–157, 2005.

[Bro09] John Browne. *Grassmann Algebra - Exploring extended vector algebra with Mathematica*. Quantica publishing, 0.50 edition, 2009.

[BT09] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[DAFD07] Maxime Descoteaux, Elaine Angelino, Shaun Fitzgibbons, and Rachid Deriche. Regularized, fast, and robust analytical q-ball imaging. *Magnetic Resonance in Medicine*, 58:497–510, 2007.

[DFP+09] Stanley Durrleman, Pierre Fillard, Xavier Pennec, Alain Trouvé, and Nicholas Ayache. A statistical model of white matter fiber bundles based on currents. *IPMI' 09 - Proceedings of the 21st International*

Conference on Information Processing in Medical Imaging, pages 114–125, 2009.

[dlT85] Gilles de la Tourette. Etude sur une affection nerveuse caractérisée par de l'incoordination motrice accompagnée d'écholalie et de coprolalie. *Arch. Neurol.*, 9:158–200, 1885.

[DMA97] G. Davis, S. Mallat, and M. Avellaneda. Greedy adaptive approximations. *Journal of Constructive Approximation*, 13(1):57–98, 1997.

[DPK+12] Stanley Durrleman, M. Prastawa, J. R. Korenberg, S. Joshi, and G. Gerig A. Trouvé. Topology preserving atlas construction from shape data without correspondence using sparse parameters. *Proceedings of Medical Image Computing and Computer Aided Intervention (MICCAI'12)*, 2012. to appear.

[DPTA08] Stanley Durrleman, Xavier Pennec, Alain Trouvè, and Nicholas Ayache. A forward model to build unbiased atlases from curves and surfaces. *2nd MICCAI Workshop on Mathematical Foundations of Computational Anatomy*, pages 68–79, 2008.

[DPTA09] Stanley Durrleman, Xavier Pennec, Alain Trouvé, and Nicholas Ayache. Statistical models of sets of curves and surfaces based on currents. *Medical Image Analysis*, 13(5):793–808, 2009.

[dR73] G. de Rham. *Variétés Différentiables, Actualites Scientifiques et Industrielles*. Hermann, 1973.

[dRnIN12] Institut Fédératif de Recherche n^o 49: Imagerie Neurofonctionelle. Brainvisa, 08-06-2012.

[Dur10] Stanley Durrleman. *Statistical models of currents for measuring the variability of anatomical curves, surfaces and their evolution*. PhD thesis, Ecole Doctorale STIC, 2010.

[fCNAAMCfBI12] Laboratory for Computational NeuroImaging Athinoula A. Martinos Center for Biomedical Imaging. Freesurfer, 08-06-2012.

[FSTD99] B. Fischl, M.I. Sereno, R.B. Tootell, and A.M. Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human Brain Mapping*, 8:272–284, 1999.

[Gla05] Joan Glaunes. *Transport par difféomorphismes de points, de mesures et de courants pour la comparaison de formes et l'anatomie numérique*. PhD thesis, Université Paris 13, 2005.

[GM98] Ulf Grenander and Michael I. Miller. Computational anatomy: an emerging discipline. *Quart. Appl. Math.*, 56:617–694, 1998.

[GPRM89] Wayne K Goodman, Lawrence H Price, Steven A Rasmussen, and Carolyn Mazure. The yale-brown obsessive compulsive scale, i. development, use, and reliability. *Arch Gen Psychiatry*, 46:1006–1011, 1989.

[GTY04] Joan Glaunes, Alain Trouvé, and Laurent Younes. Diffeomorphic matching of distributions: A new approach for unlabelled point-sets and sub-manifolds matching. *CVPR'04 Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition*, pages 712–718, 2004.

[HSL+10] K. F. Heise, B. Steven, G. Liuzzi, G. Thomalla, M. Jonas, K. Mller-Vahl, P. Sauseng, A. Mnchau, C. Gerloff, and F. C. Hummel. Altered modulation of intracortical excitability during movement preparation in gilles de la tourette syndrome. *Brain*, 133:580–590, 2010.

[JM00] Sarang C. Joshi and Michael I. Miller. Landmark matching via large deformation diffeomorphism. *IEEE Transactions on Image Processing*, 9(8):1357–1370, 2000.

[JV05] Bing Jian and Baba C. Vemuri. A robust algorithm for point set registration using mixture of gaussians. *Tenth IEEE International Conference on Computer Vision (ICCV'05)*, 2:1246–1251, 2005.

[JV11] Bing Jian and Baba C. Vemuri. Robust point set registration using gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1633–1645, 2011.

[KHB+10] Paul Krack, Marwan I. Hariz, Christelle Baunez, Jorge Guridi, and Jose A. Obeso. Deep brain stimulation: from neurology to psychiatry? *Trends in Neuroscience*, 33(10):474–484, 2010.

[Kim10] Thomas E. Kimber. An update on tourette syndrome. *Curr. Neurol. Neurosci. Rep.*, 10:286–291, 2010.

[Lec02] James F Leckman. Tourette's syndrome. *The Lancet*, 360:1577–1586, 2002.

[LRH+89] J F Leckman, M A Riddle, M T Hardin, S I Ort, K L Swartz, J Stevenson, and D J Cohen. The yale global tic severity scale: Initial testing of a clinician-rated scale of tic severity. *J Am Acad Child Adolesc Psychiatry*, 28(4):566–573, 1989.

[LZV+98] James F Leckman, Heping Zhang, Amy Vitale, Fatima Lahnin, Kimberly Lynch, Colin Bondi, Young-Shin Kim, and Bradley S. Peterson. Course of tic severity in tourette syndrome: the first two decades. *Pediatrics*, 102:14–19, 1998.

[Mil04] Michael I. Miller. Computational anatomy: shape, growth, and athropy comparison via diffeomorphisms. *NeuroImage*, 23:19–33, 2004.

[MK11] Linda Marrakchi-Kacem. *Connectivité anatomique des noyaux gris centraux: développements méthodologiques et application aux troubles moteurs*. PhD thesis, Université Paris-Sud 11, 2011.

[MKDG+12] Linda Marrakchi-Kacem, Christine Delmaire, Pamela Guevara, Fabrice Poupon, Jérome Yelnik, Sophie Lecomte, Pauline Roca, Alan Tucholka, Alexandra Durr, Jean-Francois Mangin, Marie Chupin, Stéphane Lehèricy, and Cyril Poupon. Basal ganglia connectivity inferred from tractography: dealing with direct and indirect connections. *ISMRM - 20th annual meeting and Exhibition*, 5-11 May 2012. Melbourne - Australia.

[MM11] Kevin St. P. McNaught and Jonathan W. Mink. Advances in understanding and treatment of tourette syndrome. *Nature Reviews - Neurology*, 7:667–676, 2011.

[MMTY08] Jun Ma, Michael I. Miller, Alain Trouvé, and Laurent Younes. Bayesian template estimation in computational anatomy. *NeuroImage*, 42:252–261, 2008.

[MQ09] Michael I. Miller and Anqi Qiu. The emerging discipline of computational functional anatomy. *NeuroImage*, 45:16–39, 2009.

[MTY06] Michael I. Miller, Alain Trouvé, and Laurent Younes. Geodesic shooting for computational anatomy. *Journal of Mathematical Imaging and Vision*, 24(2):209–228, 2006.

[Nes83] Y. E. Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269:543–547, 1983.

[PPC+05] M. Perrin, C. Poupon, Y. Cointepas, B. Rieul, N. Golestani, C. Pallier, D. Riviere, A. Constantinesco, D. LeBihan, and J. F. Mangin. Fiber tracking in q-ball fields using regularized particle trajectories. *Inf Process Med Imaging*, 19:52–63, 2005.

[Rie09] F. Riesz. Sur les opérations fonctionnelles linéaires. *C. R. Acad. Sci. Paris*, 149:974–977, 1909.

[SBR05] J S Stern, S Burza, and M M Robertson. Gilles de la tourette's syndrome and its impact in the uk. *Postgrad Med J*, 81:12–19, 2005.

[Sch66] Laurent Schwartz. *Théorie des Distributions*. Hermann, 1966.

[SGG00] Pamela W. Schaefer, P. Ellen Grant, and R. Gilberto Gonzalez. Diffusion-weighted mr imaging of the brain. *Radiology*, 217:331–345, 2000.

[Sin05] Harvey S. Singer. Tourette's syndrome: from behaviour to biology. *Lancet Neurol*, 4:149–159, 2005.

[SRA+04] Ziad S. Saad, Richard C. Reynolds, Brenna Argall, Shruti Japee, and Robert W. Cox. Suma: An interface for surface-based intra-and inter-subject analysis with afni. *ISBI-IEEE*, 27:1510–1511, 2004.

[ST65] E. O. Stejskal and J. E. Tanner. Spin diffusion measurements: Spin echoes in the presence of a time-dependent field gradient. *The Journal of Chemical Physics*, 42(1):288–292, 1965.

[TM09] Kamath Tallur and Robert A. Minns. Tourette's syndrome. *Paediatrics and Child Health*, 20(2):88–93, 2009.

[Tro95] Alain Trouvé. An approach of pattern recognition through infinite dimensional group actions. *Technical report*, 1995.

[VG05] Marc Vaillant and Joan Glaunès. Surface matching via currents. *Proceedings of Information Processing in Medical Imaging*, 3565:381–392, 2005.

[WGH+10] Yulia Worbe, Emilie Gerardin, Andreas Hartmann, Romain Valabrégue, Marie Chupin, Léon Tremblay, Marie Vidailhet, Olivier Colliot, and Stéphane Lehéricy. Distinct structural changes underpin clinical phenotypes in patients with gilles de la tourette syndrome. *Brain*, 133:3649–3660, 2010.

[YGSM10] Allen Yang, Arvind Ganesh, Shankar Sastry, and Yi Ma. Fast l1-minimization algorithms and an application in robust face recognition: A review. Technical report, University of California at Berkeley, 2010.