

ANALYSIS OF 3D MICROSCOPIC BRAIN IMAGES AT HIGH RESOLUTION

PAUL BERTIN

MATHEMATIQUES VISION APPRENTISSAGE
ECOLE NORMALE SUPERIEURE PARIS-SACLAY

Research Internship Report

Paris, France

August 2018

Supervisors :

STANLEY DURRLEMAN

NICOLAS RENIER

BENJAMIN CHARLIER



Informations

Intern : Paul BERTIN

E-mail : paul.f.bertin@gmail.com

Mention : Master 2

Option : Mathématiques Vision Apprentissage

Advisors : Stanley DURRLEMAN, Nicolas RENIER and Benjamin CHARLIER

Dates of the Internship : April 9 2018 - August 10 2018

Laboratory : ARAMIS Team and RENIER Team

Brain and Spine Institute
47, bd de l'hôpital 75013 PARIS
FRANCE

Host Organization : Inria Paris

Administrative Institution : Ecole polytechnique

Referring Institution : Ecole Normale Supérieure Paris-Saclay

Acknowledgments

I would like to thank my advisors Stanley DURRLEMAN, Nicolas RENIER and Benjamin CHARLIER for their time and guidance.

I also thank Sôphie SKRIABINE for very interesting conversations and cooperation on the segmentation project.

I want to especially thank Alexandre BÔNE who helped me on the *Deformetrica* code and kindly answered many questions.

I also thank Pierre ROUSSILLON who took time to explain me some aspects of his thesis on Normal Cycles.

Thanks go to to Flôrine VERNY who made me an outstanding tour of the animal facility.

Last but not least, I also thank all the members of the ARAMIS and RENIER Teams for the good atmosphere.

Abstract

During those four months in the Brain and Spine Institute under the supervision of Stanley DURRLEMAN, Nicolas RENIER and Benjamin CHARLIER, I had the opportunity to work at the interface between Mathematics and Biology Teams, to discover some aspects of Shape Analysis and Brain Vasculature Anatomy, and broadly speaking, to experience Academic Research.

The goal was to continue and extend the project *ClearVessel* initiated by Christoph KIRST and Nicolas RENIER. Based on 3D microscopic images at high resolution, we aimed to get a better understanding of the organization of the vasculature in the adult mouse brain. For this purpose, we designed and implemented various methods, heavily relying on 3D embedded graphs extracted with *ClearVessel*.

Our work consisted in developing visualization tools, performing statistical analysis, developing two distances on graphs, and trying to improve the skeletonization process in the *ClearVessel* pipeline. The two main contributions are the implementation of varifold and normal cycle distances for protocol comparison, and the implementation of a feature-based Wasserstein distance in order to compare the organization of the vasculature in different regions of the brain.

In the course of this internship, I faced difficulties related to either the manipulation of Gigabyte-big data, (multiprocessing, server management), reverse engineering of code, and dealing with multidisciplinary expectations.

Contents

Abstract

1 Biological context	2
1.1 Density and distribution of the vasculature in the brain	2
1.1.1 Gross structure of the cortical vascular system	3
1.1.2 Composition of vessels	3
1.2 Imaging techniques	3
1.3 Atlas annotation	4
2 Technical Description	6
2.1 Visualization	6
2.2 Graph	7
3 Shape analysis	8
3.1 Context	8
3.1.1 Preliminaries	8
3.1.2 Varifolds	8
3.1.3 RKHS of oriented varifolds	9
3.1.4 Normal cycles	10
3.1.5 Local distances	11
3.2 Protocol comparison	12
3.2.1 Implementation	12
3.2.2 Results	12
3.2.3 Possible extensions	12
4 Organization analysis	14
4.1 Organization analysis and Optimal Transport	14
4.1.1 Exact computation	14
4.1.1.1 The exact transportation problem	14
4.1.1.2 Formulation as a linear problem	15
4.1.2 Approximate computation	16
4.2 Data exploration and features	16
4.2.1 Spatial analysis	16
4.2.1.1 Spatial features	17
4.2.1.2 Discriminative power of spatial features	17
4.2.2 Analysis of Vessels	18

4.2.2.1	Vessel features	19
4.2.2.2	T-SNE visualization	19
4.3	Organization of the vasculature accross the brain	19
4.4	Future possible extensions	21
4.4.1	Short term	21
4.4.2	Long term	24
5	Vessel binarization	25
5.1	Deep Learning	25
5.1.1	Origin and understanding of Deep Learning	25
5.1.1.1	Kernel classifiers	26
5.1.1.2	Convolutional networks	27
5.1.2	Autoencoders	27
5.2	Binarization	27
5.2.1	Architecture	28
5.2.2	Optimization	29
5.2.2.1	Optimization parameters	29
5.2.2.2	Regularization	29
5.2.3	Possible extensions	30
A	List of Brain regions by cluster	32

Introduction

As part of the master *Mathématiques Vision Apprentissage* of École Normale Supérieure Paris Saclay, I did a four month research internship at the Brain and Spine Institute located in Paris. I was part of both the ARAMIS Team (Inria Paris) which designs new approaches to study the structural and functional aspects of the brain, and the RENIER Team which studies brain plasticity. Thus I had the chance to study in a multidisciplinary environment and got advices from both biologists and mathematicians. Along this internship, I was supervised by Stanley DURRLEMAN, Nicolas RENIER and Benjamin CHARLIER.

The purpose of this internship was to get a first experience of research and to put into practice the tools studied during the academic year. I have been interested in computational neurosciences for quite a long time, and applying mathematical tools to biological data in order to better understand the mechanisms of the brain was a very interesting opportunity.

The RENIER Team has developed a new technique for whole brain immunostaining and volume imaging. Together with Christoph KIRST they developed the *TopoVessel* project to efficiently extract 3D embedded graphs from the volume images of the vasculature. The goal of my intership was to develop statistical tools and shape analysis techniques in order to get a better insight into this gigantic amount of data.

In this report, I will describe the context of the project, from both the biological and mathematical points of view, present some aspects of my code and possible extensions, and then my results.

Chapter 1

Biological context

During this internship, we had to study both the biological and mathematical aspects of the project. Here we present the context from the biological point of view.

1.1 Density and distribution of the vasculature in the brain

Those explanations rely on a review published in *Neuroimage* in 2017 [1]. The knowledge of the cerebral vasculature is crucial to understand the principles of the blood flow in the cerebral vasculature. The blood flow is related to neural processing as it controls the energy supply.

We will focus especially on the adult mouse brain, as this corresponds to the images we were provided with.

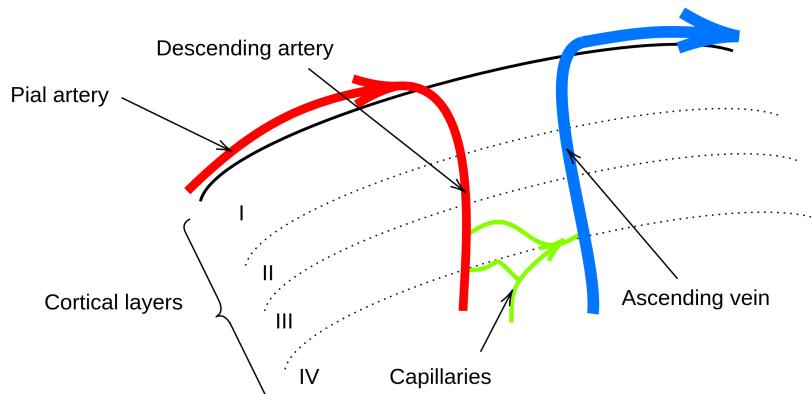


FIGURE 1.1: Schematic view of the vascular structure in the cortex

1.1.1 Gross structure of the cortical vascular system

The cerebral arteries run along the surface of the cortex and ramify into a complex network of pial arteries. Then the arteries leave the pial network and penetrate perpendicularly into the cortex and form descending arteries.

The cortical arteries send off collaterals at different cortical depths. There exists several types of arteries. Some of them feed superficial cortical layers, whereas others feed deeper layers. Some cortical arteries even penetrate the entire cortex without any collateral until reaching white matter.

Then the cortical arteries diverge into arterioles and finally end in the capillary network. This is at the level of capillaries that most of the exchange of energy and oxygen occurs. Capillaries then converge to the venous system, in which main cortical veins are oriented perpendicularly to the cortical surface.

The main scheme of cortical blood flow is thus a descending flow from the surface to some specific depth, an exchange of energy and oxygen in the capillary network and an ascending flow to transport away the blood through ascending veins.

The microvascular (or capillary) system is a redundant network with a mesh width of approximately $50\mu m$. This mesh width is probably related to the diffusion constant of oxygen in the brain tissue.

Those observations give us clues to analyses the vessel organization on a bigger scale.

1.1.2 Composition of vessels

Arteries, capillaries and veins are all composed of an endothelial cell layer (forming the blood-brain barrier) and a thin basal membrane.

Capillaries only consist of these two elements, but arteries (and in a lesser extent veins) are covered with a muscle sheath. Those muscles are responsible for the regulation of the blood flow by changing the vessel diameter.

This specific composition has been used by the RENIER Team in order to immunolabel specifically arteries.

1.2 Imaging techniques

Until the work done by the RENIER Team, the brain vasculature had not been observed completely. Previous techniques either focused on a small region of the brain (of the

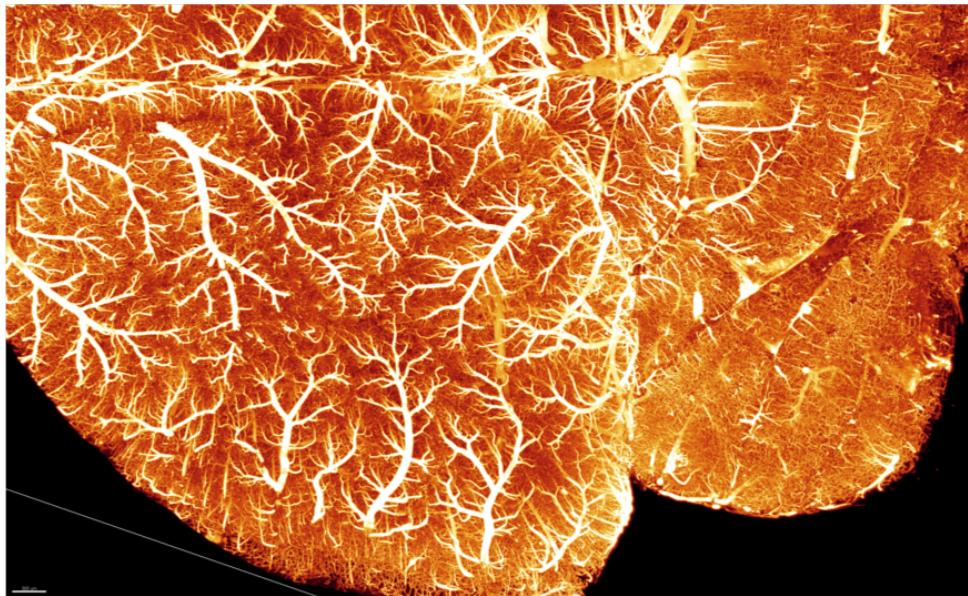


FIGURE 1.2: 3D microscopic image of the brain vasculature on a transverse projection

order of magnitude of a μm^3) or scaled to the whole brain without capturing the capillary structure.

The protocol developed by the RENIER Team allowed us for the first time to obtain the full brain vasculature with a high level of details. The blood and the muscle sheath are immunolabeled after a transparisation of the brain. Then, the acquisition is made using a light-sheet microscope. An example is show on [Figure 1.2](#)

Final 3D images can weight up to 300 Gigabytes and analysing them is obviously a computational and mathematical challenge. Together with Christoph KIRST, they developed *ClearVessel* in order to extract 3D embedded graphs from those images. Those graphs typically had about 90 million vertices.

1.3 Atlas annotation

Biologists have developed an atlas annotation of the adult mouse brain available online at <http://atlas.brain-map.org/>. The *ClearVessel* pipeline allows to register brains on this annotation and thus specific subgraphs corresponding to specific regions of the brain can be extracted.

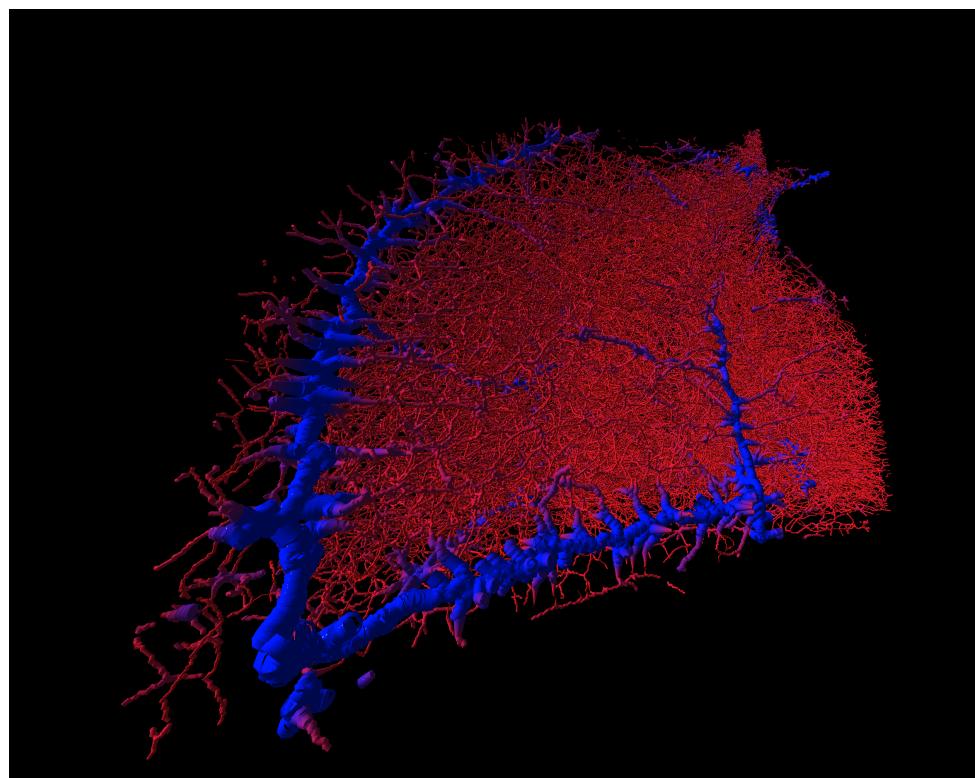


FIGURE 1.3: Graph of the *Central Lobule* region of the brain. Big vessels are blue and capillaries are red

Chapter 2

Technical Description

We will briefly describe some aspects of our code which is available at <https://gitlab.icm-institute.org/paul.bertin/TopoVessel>.

2.1 Visualization

In order to visualize graphs embedded in 3D with the radius information, we developed a 3D visualization tool based on the *vispy* library to plot vessels as tubes.

To do so, we created a class *VesselVisual* which wraps the class *MeshVisual* from the *vispy* library. Given the vertex positions, edges and radii of the graph, this class constructs a mesh object with all surfaces which will be plotted.

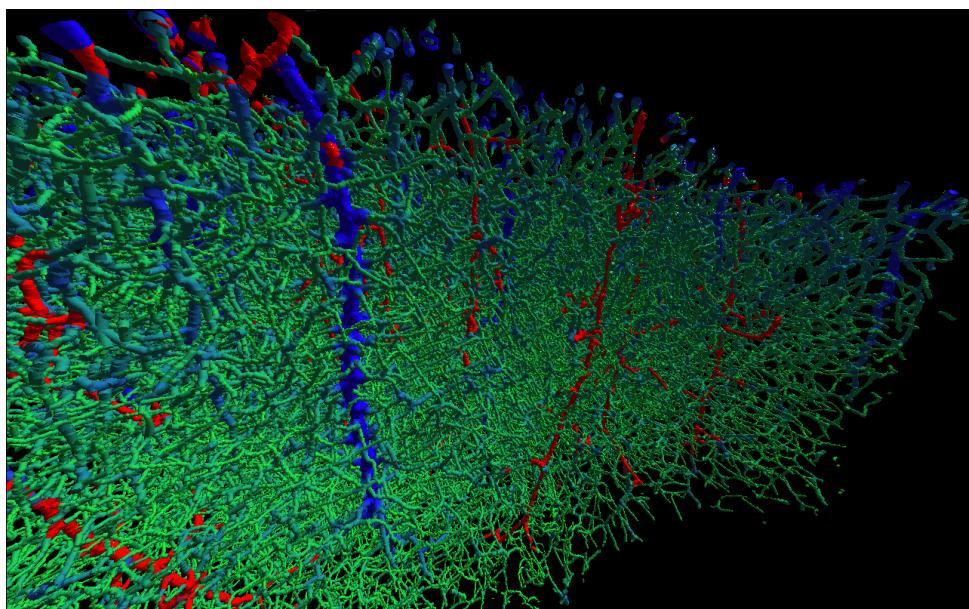


FIGURE 2.1: Descending arteries (red) and ascending veins (blue) in the barrel cortex

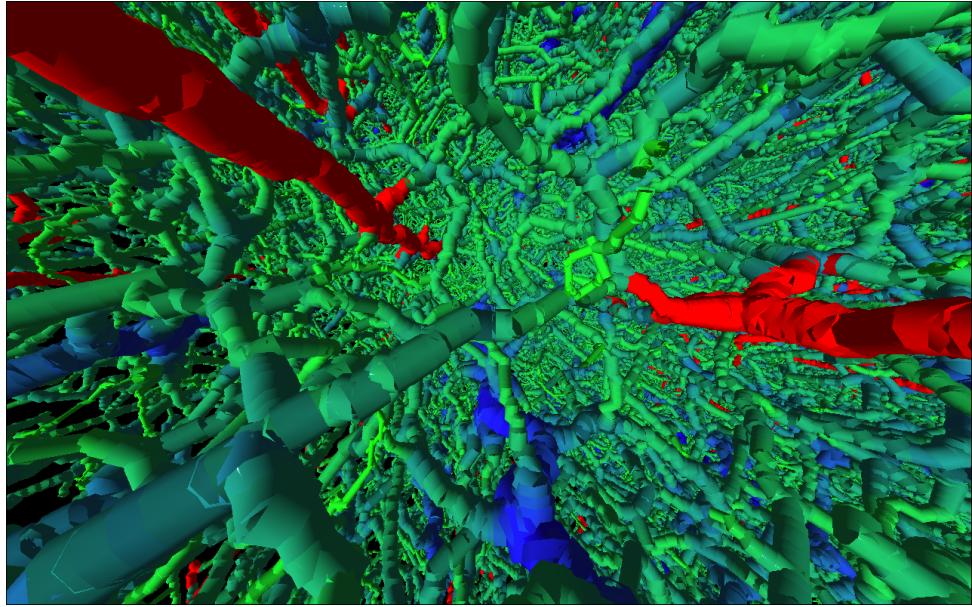


FIGURE 2.2: Descending arteries (red) in the barrel cortex

We use the methods from *vispy* to plot in 3D. Unfortunately, the precomputation (necessary to render the final plot without lag) is very slow and not parallelized. Up to now, the final mesh is saved as basic numpy arrays. It would be great to be able to save them in a *Paraview* compatible format, in order to take advantage of the optimized tools of *Paraview*.

Our plot functions take colormap functions as argument. This enables the user to easily change the colors of the visualization to see whatever characteristic he is interested in.

For more details, please refer to the *Visualization.visualize.py* file.

2.2 Graph

As in the *ClearVessel* project developed by Christoph KIRST, we used the *graph-tools* library which is convenient for computations on very large graphs.

Not that we call a vessel a part of a capillary (or artery or vein) which is between two intersections.

ClearVessel had a graph reduction method. Reducing a graph means merging all the edges of a vessel into one. This is useful to run algorithms more efficiently. We extended this method to manage graph annotations (for arteries, or other vessel features) and to be able to extract subgraphs in a coherent manner.

Chapter 3

Shape analysis

In order to study the organization of the vasculature, we had to explore several tools and mathematical frameworks. As we had access to graphs embedded in 3D space, we dived into methods of shape analysis which have been developed in the past decades.

3.1 Context

Here we present the frameworks of varifolds and normal cycles, for which we show application results later in the text. Those frameworks allow us to compute distances on graphs with lots of relevant properties for our application.

3.1.1 Preliminaries

Thoses explanations rely on a paper from I. Kaltenmark, B. Charlier and N. Charon. [2]

A shape usually refers to a graph or a mesh embedded in ambient space \mathbb{R}^n with $n = 2$ or $n = 3$. For modeling purposes, we will now adopt a broader view (which includes those specific cases) and see shapes as smooth submanifolds or reunion of smooth submanifolds of dimension 1 (lines) or 2 (surfaces) embedded in \mathbb{R}^n .

3.1.2 Varifolds

Let X be a smooth submanifold with finite total volume $vol(X) < \infty$. We can associate to X an *oriented varifold* μ_X which is a distribution on the product space $\mathbb{R}^n \times \mathbb{S}^{n-1}$. Here \mathbb{S}^{n-1} refers to the unit sphere (as it can be identified with the space of tangent orientations

for shapes of dimension or codimension 1). The measure μ_X is defined as follows :

$$\mu_X = \int_X \omega(x, \vec{t}(x)) d\text{vol}(x) \quad (3.1)$$

for any smooth test function $\omega : \mathbb{R}^n \times \mathbb{S}^{n-1} \rightarrow \mathbb{R}$.

In the distribution sense, we may write $\mu_X = \int_X \delta_{(x, \vec{t}(x))} d\text{vol}(x)$ with the Dirac delta $\delta_{(x, \vec{t}(x))}(\omega) \triangleq \omega(x, \vec{t}(x))$. This ensures that the identification $X \rightarrow \mu_X$ is an injection. Note that for usual graphs and meshes, we will choose a discretization scheme so that we will only deal with finite sums of Dirac deltas.

In the case of a graph G embedded in 3D space without vertex of degree 0, we can consider that $G = \bigcup_i v_i$ where v_i is the i^{th} edge. In this case, $\mu_G = \sum_i \mu_{v_i}$ where μ_{v_i} is the oriented varifold associated with the edge v_i which is approximately $\mu_{v_i} = l_i \delta_{(x_i, \vec{t}_i, r_i)}$ with l_i the length of the edge, r_i its radius and x_i the center of the edge.

Note that here we made a slight extension of the previous model by taking μ_G as a distribution over $\mathbb{R}^3 \times \mathbb{S}^2 \times \mathbb{R}$ in order to take into account the radius of a given edge.

3.1.3 RKHS of oriented varifolds

The idea here is to take a specific family of test functions ω for our varifold representation. We will take test functions belonging to a Reproducing Kernel Hilbert Space W with an associated kernel of the form $k_{pos} \otimes k_{or} \otimes k_{rad}$. This is a convenient choice as the combination of effects due to position, orientation and radius will be easily interpretable.

One can show that (see [2]) the following expression defines a scalar product on shapes :

$$\langle X, Y \rangle = \langle \mu_X, \mu_Y \rangle_{W^*} \quad (3.2)$$

$$\triangleq \int_{X \times Y} k_{pos}(x, y) k_{or}(\vec{t}(x), \vec{t}(y)) k_{rad}(r(x), r(y)) d\text{vol}^X(x) d\text{vol}^Y(y) \quad (3.3)$$

where X and Y are smooth submanifolds with finite total volume.

We then have to choose specific kernels which will induce convenient properties for our application. We will thus choose *Gaussian* kernels for the position and the radius, and *Binet* kernel for the orientation. This setting corresponds to the functional unoriented varifold.

Note that choosing the linear kernel for the orientation would correspond to currents.

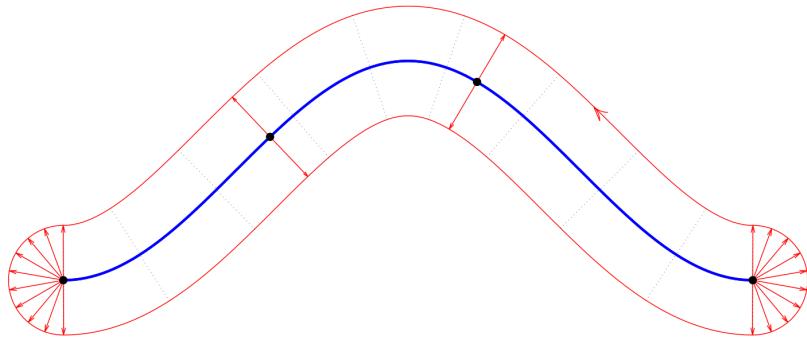


FIGURE 3.1: A planar curve in blue and its unit normal bundle in red. Source [3]

3.1.4 Normal cycles

Normal cycles are a variation on the previous approach. Compared to varifolds, they give more importance to the extremities of the shape. This is very interesting as we are interested in detecting interrupted vessels for example (which would give two extremities instead of zero).

Formally, the normal cycle of a shape is the current associated with its normal bundle. Those explanations rely on the thesis of Pierre ROUSSILLON [3].

Definition 3.1. The normal cone of a shape X at a point x is

$$\text{Nor}(X, x) \triangleq \{u \in \mathbb{R}^d, \forall v \in T_x X, \langle u, v \rangle \leq 0\} \quad (3.4)$$

Definition 3.2. The unit normal bundle of X is

$$\mathcal{N}_X \triangleq \{(x, n) \in \mathbb{R}^d \times \mathbb{S}^{d-1}, x \in X, n \in \text{Nor}^u(X, x)\} \quad (3.5)$$

where $\text{Nor}^u(X, x) \triangleq \text{Nor}(X, x) \cap \mathbb{S}^{d-1}$

Definition 3.3. $\langle \cdot | \cdot \rangle$ refers to the duality product on $\Lambda^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1})$, the space of $(d-1)$ vectors of $\mathbb{R}^d \times \mathbb{S}^{d-1}$ [3, pages 51-52]. The algebraic dual of $\Lambda^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1})$ is the space of $(d-1)$ forms $\Lambda_{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1})$. Thus :

$$\langle \cdot | \cdot \rangle : \Lambda_{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1}) \times \Lambda^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1}) \rightarrow \mathbb{R} \quad (3.6)$$

$$\phi, x \mapsto \phi(x) \quad (3.7)$$

Definition 3.4. The Normal Cycle of a set X is :

$$N(X)(\omega) \triangleq \int_{\mathcal{N}_X} \langle \omega(x, n) | \tau_{\mathcal{N}_X}(x, n) \rangle d\text{vol}(x, n) \quad (3.8)$$

where ω is a differentiable form and $\tau_{\mathcal{N}_X}(x, n)$ is the $(d-1)$ vector associated with an orthonormal positively oriented basis of $T_{(x,n)}\mathcal{N}_X$.

In the same manner as before, we then choose ω to belong to some RKHS to derive suitable metrics on shapes.

In the case of graphs and choosing the linear kernel for normals, the approximated scalar product between Normal Cycles is [3] :

$$\langle N(G_1), N(G_2) \rangle_{W^*} = \frac{\pi^2}{2} \sum_{x,y} k_{pos}(x, y) l_x l_y \cos^2(\theta_{x,y}) \quad (3.9)$$

$$+ \frac{16\pi^2}{3} \sum_{c,s} k_{pos}(c, s) \left(1 - \frac{\deg(c)}{2}\right) \left(1 - \frac{\deg(s)}{2}\right) \quad (3.10)$$

where the first sum is over edges and the second over vertices. $\theta_{x,y} = \langle x/l_x, y/l_y \rangle$.

3.1.5 Local distances

For interpretability purposes, we are interested in developing tools capturing local differences between shapes.

One way to do this would be to take the norm of the gradient of the distance with respect to each vertex in the graphs. We would then have for each vertex in our graphs, an indication of its *proximity* with the other graph.

Another technique, is to filter our shape with multiple localized windows and then compute the distances described above on the filtered shapes.

In practice we used *Gaussian* windows.

Definition 3.5. Let GF_c be the *Gaussian* window centered in $c \in \mathbb{R}^3$. Filtering the graph $G = \sum l_i \delta_{x_i, \vec{t}(x_i)}$ gives :

$$GF_c(G) \triangleq \sum e^{-\frac{\|x_i - c\|^2}{\sigma^2}} l_i \delta_{x_i, \vec{t}(x_i)} \quad (3.11)$$

Our filters are linear bijections and thus for any center of window c , $\langle GF_c(G_1), GF_c(G_2) \rangle = \langle \mu_{GF_c(G_1)}, \mu_{GF_c(G_2)} \rangle_{W^*}$ is still a scalar product on graphs.

We can then visualize the function $c \in \mathbb{R}^3 \rightarrow \|GF_c(G_1), GF_c(G_2)\|$ to understand where graphs are different from one another.

One could also use windows with compact support for scalability : the computational cost of computing one local distance would then be independent of the total size of the graph. A good choice would probably be windows associated with a truncated *Gaussian* function (of the form $x \rightarrow \mathbb{1}_{I_c} e^{-\frac{\|x - c\|^2}{\sigma^2}}$ where I_c is an interval of \mathbb{R}^3 centered in c). In any

case, one should avoid to use functions with *big* discontinuities such as indicator functions to avoid side effects.

3.2 Protocol comparison

In order to compare the graphs obtained with the PODO protocol (corresponding to immunolabeling the membrane of the vessels with Podocalyxin) and the IgG-protocol (corresponding to immunolabeling directly the blood with Immunoglobulin G), we computed local distances (with either varifolds or normal cycles) as explained in Section 3.1.5.

3.2.1 Implementation

We implemented a functional varifold distance and a normal cycle distance (as well as their local variants using Gaussian filters) using the *pytorch* library. The use of *pytorch* was convenient as similar code was available in the *Deformetrica* software, and the auto-differentiation would be useful in case one wants to use gradients in the future.

3.2.2 Results

We then visualize local varifold distances projected on graphs as shown on Figure 3.2. Intuitively, local distances seem to make sense, as portions where the reconstructed graphs are similar are close to each other while portions where reconstructed graphs are very different (*e.g.* a vessel appears in one graph and not in the other) are distant.

The visualization of local normal cycle distances proved to give very similar results to varifold distances. It might be because, in the computation of the distance, the term on edges is predominant over the term on vertices. Further analysis should be performed to confirm this hypothesis.

3.2.3 Possible extensions

This tool has been designed to assist biologists to spot reconstruction errors while visualizing graphs. Thus the colormap directly plotted on the graphs appears to be adapted. If one wants to use this tool to automatically assess that graphs correspond to the *same* reconstruction, one should think about using this distance computation in a pipeline to indicate coordinates where problems happen.

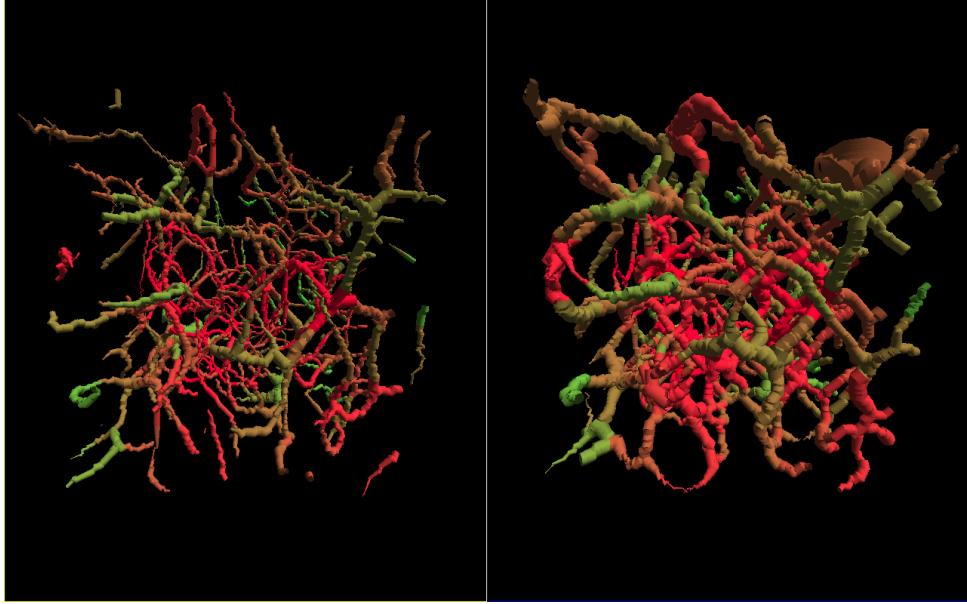


FIGURE 3.2: Local varifold distances projected with a colormap on graphs. Red corresponds to big distances and green to small ones. Left is the graph reconstructed with the IGG protocol (blood immunolabeling) and right is the PODO protocol (vessel immunolabeling)

Note that graph reconstructions are still very sensitive to the parameters used for the *ClearVessel* pipeline, and this can be an issue if one wants to compare the accuracy of the protocols themselves (and not the accuracy of the parameters used). More specifically, the two images from the two protocols might have different brightness and contrast, which requires to use different parameters for the *ClearVessel* pipeline. How to assess that an artifact in one of the graphs is due to limitations inherent to the protocol and not to a bad choice of parameters ?

Chapter 4

Organization analysis

Shape analysis frameworks define topologies which are well suited tools when one wants to compare graphs which are *very* similar, that is to say graphs which have similar spatial conformations.

Nevertheless, we are also interested in finding other representations of graphs so that it is possible to compare graph organizations without being blinded by the exact spatial conformation of each graph. More precisely, we would like to compute a distance between graphs corresponding to different regions of the brain in a way that reflects the different organizations of the vasculature.

4.1 Organization analysis and Optimal Transport

We first present some elements about the computation of optimal transport.

4.1.1 Exact computation

4.1.1.1 The exact transportation problem

The study of Optimal Transport was first initiated by G. MONGE, in order to minimize the effort (*i.e.* the amount of work) necessary to move dirt from one place to another. The theory was then extended by KANTOROVICH whose formulation we use thereafter [4, pages 42-44].

Formally, given two distributions of mass, the goal is to find a mapping between them which is optimal according to some cost function.

Definition 4.1. Let $\mathcal{P}(\mathbb{R}^d)$ be the set of all distributions on \mathbb{R}^d . Let μ_A and $\mu_B \in \mathcal{P}(\mathbb{R}^d)$. A solution to the *Kantorovich's* problem is given by a coupling $\pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d)$ satisfying the following infimum :

$$\inf_{\pi} \left\{ \int_{\mathbb{R}^d \times \mathbb{R}^d} c(x, y) d\pi(x, y) : \pi \in \Pi(\mu_A, \mu_B) \right\} \quad (4.1)$$

where c stands for the cost function and $\Pi(\mu_A, \mu_B)$ for the set of all couplings between μ_A and μ_B i.e. distributions over $\mathbb{R}^d \times \mathbb{R}^d$ with marginals μ_A and μ_B .

We can show that for a good choice of the cost function c , the solution to *Kantorovich's* problem exists.

Lemma 4.2. Let $c : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R} \cup \infty$ be lower continuous and bounded from below. Then the infimum of *Kantorovich's* problem is always attained by a transport plan $\pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d)$.

For instance, if one chooses c to be the 1-norm or 2-norm in \mathbb{R}^d (what we use in practice), the problem admits a solution.

4.1.1.2 Formulation as a linear problem

As we will be dealing with computational applications, we will now restrict ourselves to the discretized problem, which we will formulate as a linear problem. This will enable us to use out of the box linear solvers.

Let us consider two discrete distributions $\mu_A = \sum_{i=1}^{n_A} p_{A,i} \delta_{x_{A,i}}$ and $\mu_B = \sum_{i=1}^{n_B} p_{B,i} \delta_{x_{B,i}}$ where n_A and n_B are the number of points, δ_x is the Dirac at location x and $(x_{A,i})_{i=1}^{n_A}, (x_{B,i})_{i=1}^{n_B} \in (\mathbb{R}^d)^{\mathbb{N}}$ two point clouds.

The set of couplings between μ_A and μ_B is :

$$\Pi(\mu_A, \mu_B) = \left\{ (\pi_{i,j})_{i,j} \in (\mathbb{R}^+)^{n_0 \times n_1} : \forall i, \sum_j \pi_{i,j} = p_{A,i}, \forall j, \sum_i \pi_{i,j} = p_{B,j} \right\} \quad (4.2)$$

and the KANTOROVICH's problem becomes :

$$\pi^* = \operatorname{argmin}_{\pi \in \Pi} \sum_{i,j} c_{i,j} \pi_{i,j} \quad (4.3)$$

where $c_{i,j} = \|x_{A,i} - x_{B,j}\|_p^p$ for $p = 1$ or 2 . Intuitively, $c_{i,j}$ represents the amount of work necessary to move one unit of mass from $x_{A,i}$ to $x_{B,j}$, and $\pi_{i,j}$ the amount of mass displaced between $x_{A,i}$ to $x_{B,j}$.

The L^p -Wasserstein distance can then be computed as $W_p(\mu_A, \mu_B)^p = \sum_{i,j} \pi_{i,j} c_{i,j}$.

The constraint $\pi \in \Pi$ can be rewritten in matrix form, and we thus have a linear problem which can be solved using the simplex algorithm or interior points methods.

4.1.2 Approximate computation

Solving the exact transportation problem quickly becomes intractable when dealing with large domains. To overcome this issue, one can relax the problem with an entropy regularization, which leads to a reduction of the dimension of the problem. Those methods showed impressive results when dealing with images [5] and they could enable us to use more features (thus having distributions in higher dimensional spaces) and generally, to use discretized domains with a higher number of bins.

The entropy of a coupling is given by :

$$H(\pi) \triangleq - \int_{\mathbb{R}^d \times \mathbb{R}^d} \pi(x, y) \ln(\pi(x, y)) dx dy \quad (4.4)$$

And the regularized problem is :

$$W_{p,\gamma}^p(\mu_A, \mu_B) = \inf_{\pi} \left\{ \int_{\mathbb{R}^d \times \mathbb{R}^d} c(x, y) d\pi(x, y) - \gamma H(\pi) : \pi \in \Pi(\mu_A, \mu_B) \right\} \quad (4.5)$$

with γ some positive constant. Note that here $W_{p,\gamma}(\mu_A, \mu_B)$ is only a pseudo-distance as it is not always positive.

The efficiency of the method comes from specificities of the numerical framework which we will not detail here, the reader can refer to [5]. In short, the number of unknowns in the problem is reduced from n^2 to $2n$.

4.2 Data exploration and features

4.2.1 Spatial analysis

We implemented various spatial features on graphs, *i.e* features taking advantage of the 3D embedding of our graph.

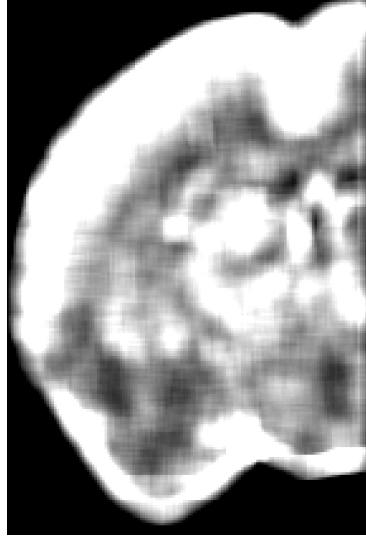


FIGURE 4.1: Density of vessels in the graph seen on a coronal section. We see a higher density in cortical regions than in most over regions of the brain

4.2.1.1 Spatial features

We adapted the code from *Clearvessel* to visualize graph spatial features as 3D images. This can give a good insight to biologists and help us to analyze the relevance of each feature. For example one can visualize the density of vessels in the brain. An example is shown on [Figure 4.1](#).

We used the following features for our experiments : density of vessels, density of intersections, biggest radius in a cubic patch, and distribution of vessels in a cubic patch. Those are simple features which may reflects simple differences and obviously other, more complex features could be used.

4.2.1.2 Discriminative power of spatial features

In order to assess the discriminative power of our feature representation. we set up a supervised pipeline to discriminate between patches extracted from different regions of the brain.

The pipeline is described on [Figure 4.2](#). We extract a subgraph (called a patch) corresponding to a small cube of the original 3D image (typically of size $200\text{px} \times 200\text{px} \times 200\text{px}$) which is then fed to a *Random Forest* classifier.

Our quantitative results for Isocortex and Hippocampus are shown in [Table 4.1](#). The different lines of the table correspond to the accuracies using only one feature. We used the random forest classifier with 10 estimators and a maximal depth of 10.

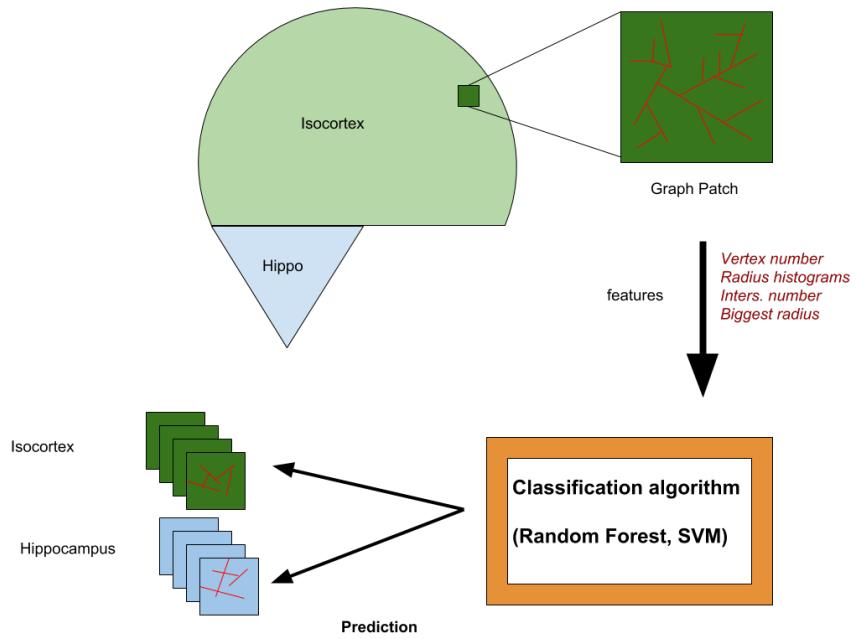


FIGURE 4.2: Description of the supervised pipeline to assess the discriminative of our spatial feature representations

	Train acc.	Test acc.	Test prec.	Test recall
N_{vertex}	0.79	0.54	0.55	0.55
$N_{intersec}$	0.72	0.58	0.59	0.58
Biggest ves.	0.80	0.53	0.53	0.53
Hist (small)	0.61	0.54	0.61	0.55
Hist (big)	0.90	0.65	0.66	0.66
all	0.91	0.70	0.71	0.70

TABLE 4.1: Discriminative power of features (Isocortex vs Hippocampus) using random forest
 $n_{estimators} = 10$, $max_{depth} = 10$ on 200×200 patches

For this two class classification task, we achieve 70% test accuracy, but the algorithm might rely mostly on artifacts inherent to the immunolabeling process (*e.g.* deeper regions have lower average radius, as explained in Section 4.2.2.2) and not on differences in the actual structure of the vasculature.

4.2.2 Analysis of Vessels

We have performed analysis on the different types of vessels in the brain. We would like to see if some specific types of vessels are specific to some regions of the brains. For example, we could expect that big long straight vessels would be found only in upper layers of the cortex (corresponding to ascending veins and descending arteries).

4.2.2.1 Vessel features

We used rather simple features for vessels, mean radius along the vessel, its length, its orientation, its curvature and its torsion. Those are features that biologists have used informally to distinguish between different types of vessels such as capillaries and arteries.

4.2.2.2 T-SNE visualization

In order to analyse the vessel feature representations and find some relevant low dimensional embedding, we used *t*-Distributed Stochastic Neighbor Embedding (*t*-SNE), which has been presented in 2008 [6].

Thus we performed T-SNE visualizations of our vessel feature representations as shown in [Figure 4.3](#) and [Figure 4.5](#). Vessels seem to organize along a one dimensional manifold which is quite surprising at first. In fact, they are organized by increasing order of radius.

This might be due to an artifact of the immunolabeling process. The immunomarker diffuses from the cortex to deeper regions in the brain. Thus, cortical regions appear brighter in the volume image and reconstructed radii are bigger. We tried to overcome this artifact by shifting radii distributions so that all regions of the brain have radii distributions with the same average radius. It turned out that the vessels were then organized by length.

Those observations point out that our feature representations do a quite bad job at extracting the relevant characteristics of the different types of vessels in the different regions of the brain. Therefore, one should try to extract more features or to make existing features more robust to noise.

4.3 Organization of the vasculature accross the brain

In order to compute a distance between graphs in a way that reflects the different organizations of the vasculature, we find a feature-based surjection of our graphs to some space of probability distribution and then compute the L_1 -Wasserstein distance in this space.

We consider regions of the brain as feature distributions. We then compute Wasserstein distances between all pairs of regions. Results are shown on [Figure 4.6](#), [Figure 4.7](#) and [Figure 4.8](#). For tractability reasons, we only used three features to represent vessels : radius, length and curvature.

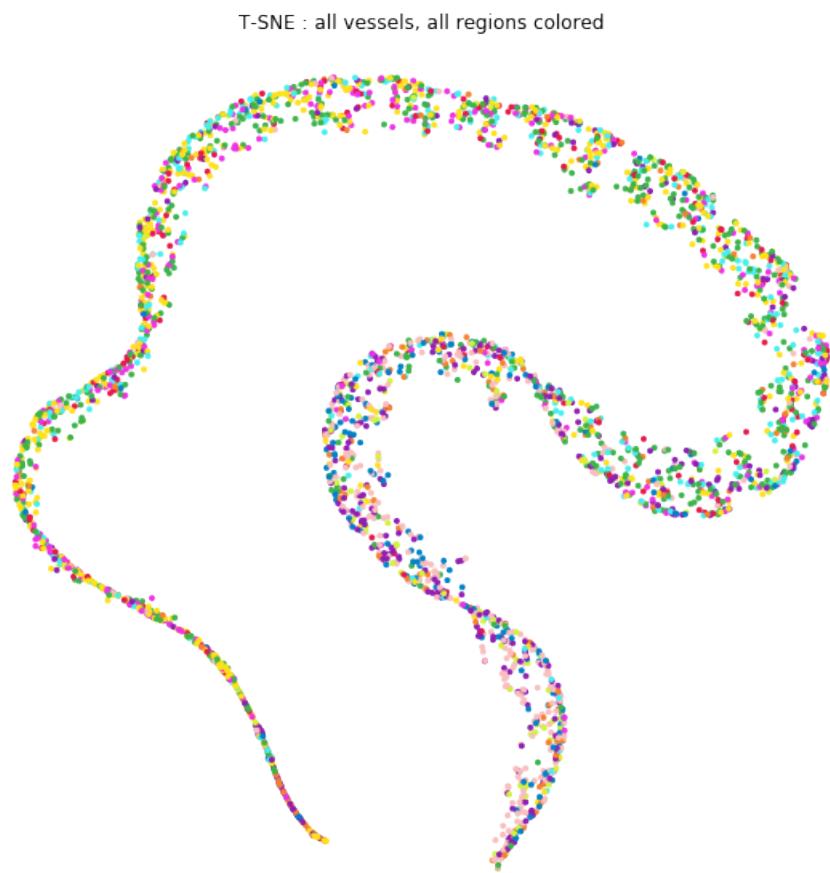


FIGURE 4.3: T-SNE visualization of the vessel feature representation with all vessels (subsampled) in ten regions of the brain

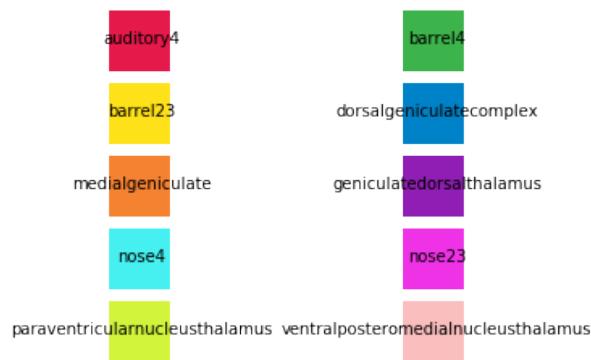


FIGURE 4.4: Legend of region colors

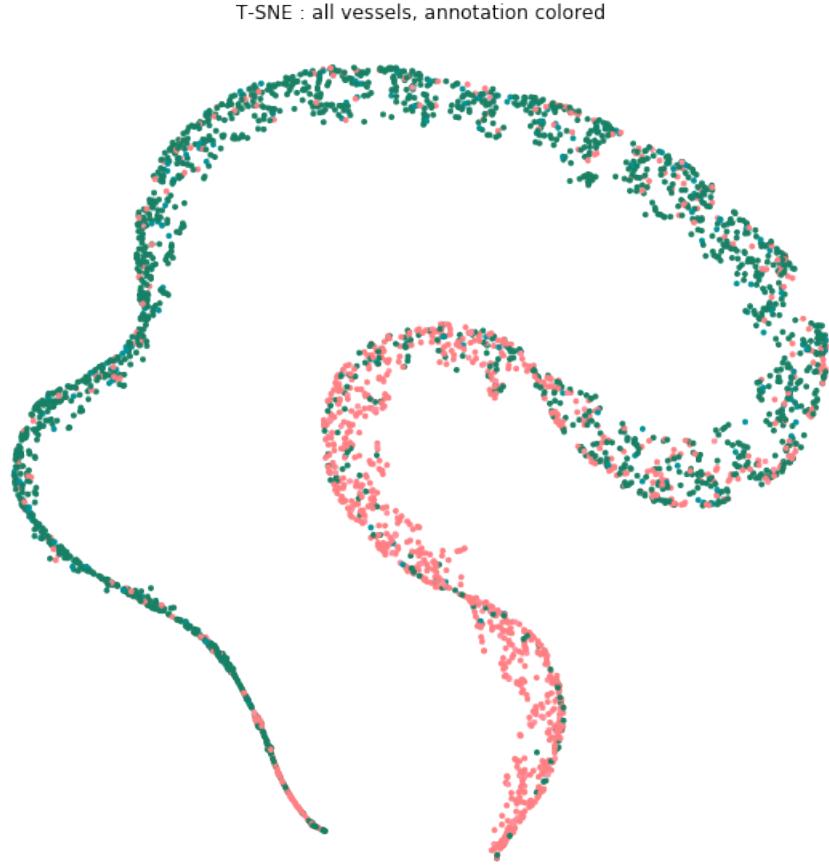


FIGURE 4.5: T-SNE visualization of the vessel feature representation with all vessels (subsampled) in ten regions of the brain. Color corresponds to atlas annotation colors

For preprocessing, we shift distributions of vessels so that all regions have the same average radius. After shifting, we only take into account vessels with radii lower than 5000 (arbitrary unit) to avoid artifacts.

We then use hierarchical clustering to cluster together regions of the brain. Those results would need more advanced analysis by biologists, but we note that the different cortical layers tend to be clustered in different clusters, while regions corresponding to the same cortical layer tend to be clustered together.

4.4 Future possible extensions

4.4.1 Short term

The *ClearVessel* project is still in progress and some issues regarding vessel interruptions, and *shadow* vessels (nonexistent vessels which are created by the *Clearvessel* pipeline

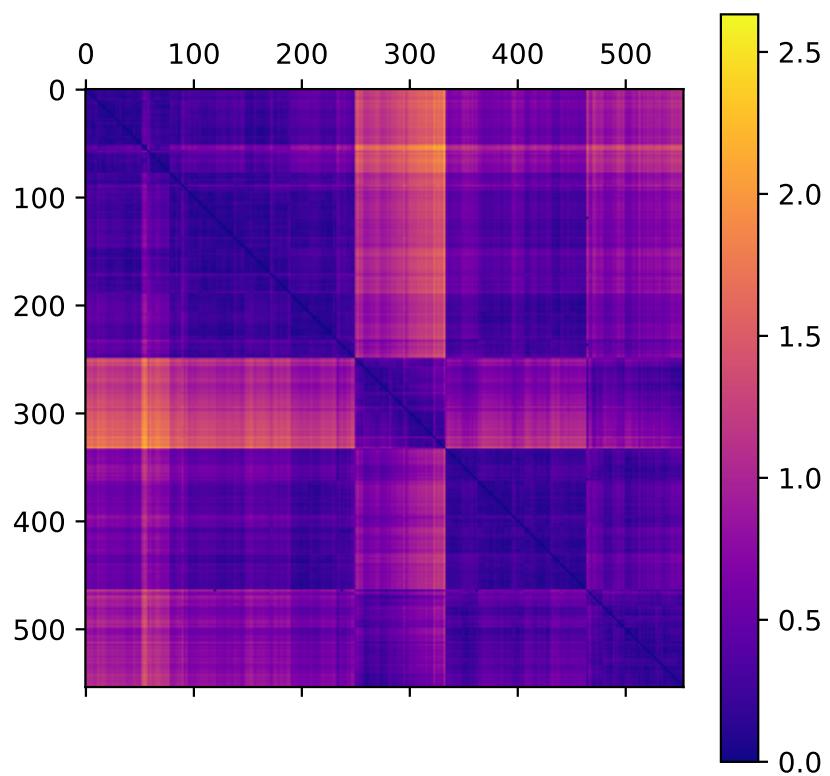


FIGURE 4.6: Distance matrix between all regions in the brain. Rows and columns are ordered according to hierarchical clustering using *Ward* criterion

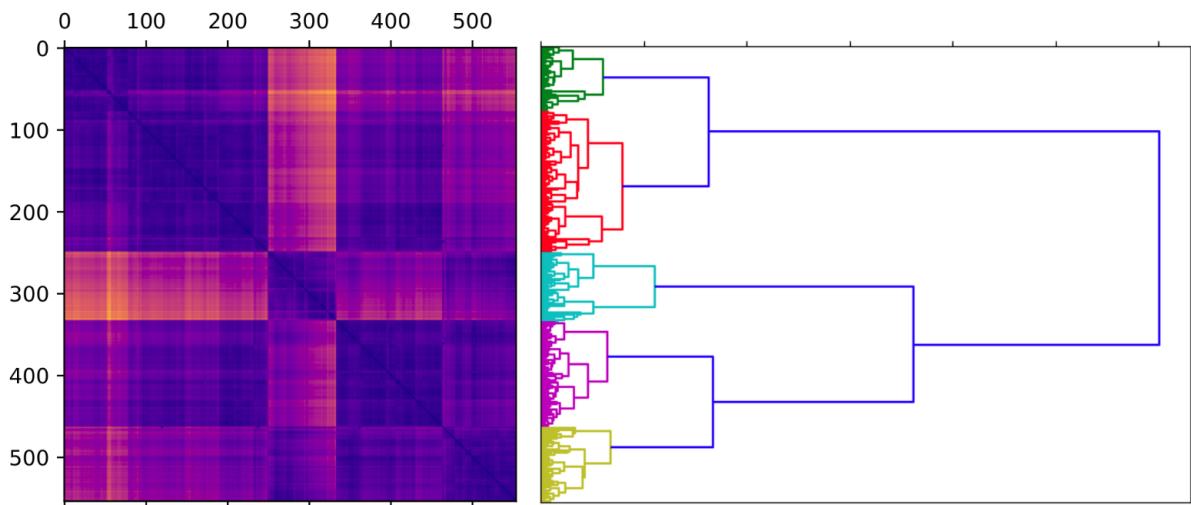


FIGURE 4.7: Distance matrix and associated dendrogram. Rows and columns are ordered according to hierarchical clustering using *Ward* criterion

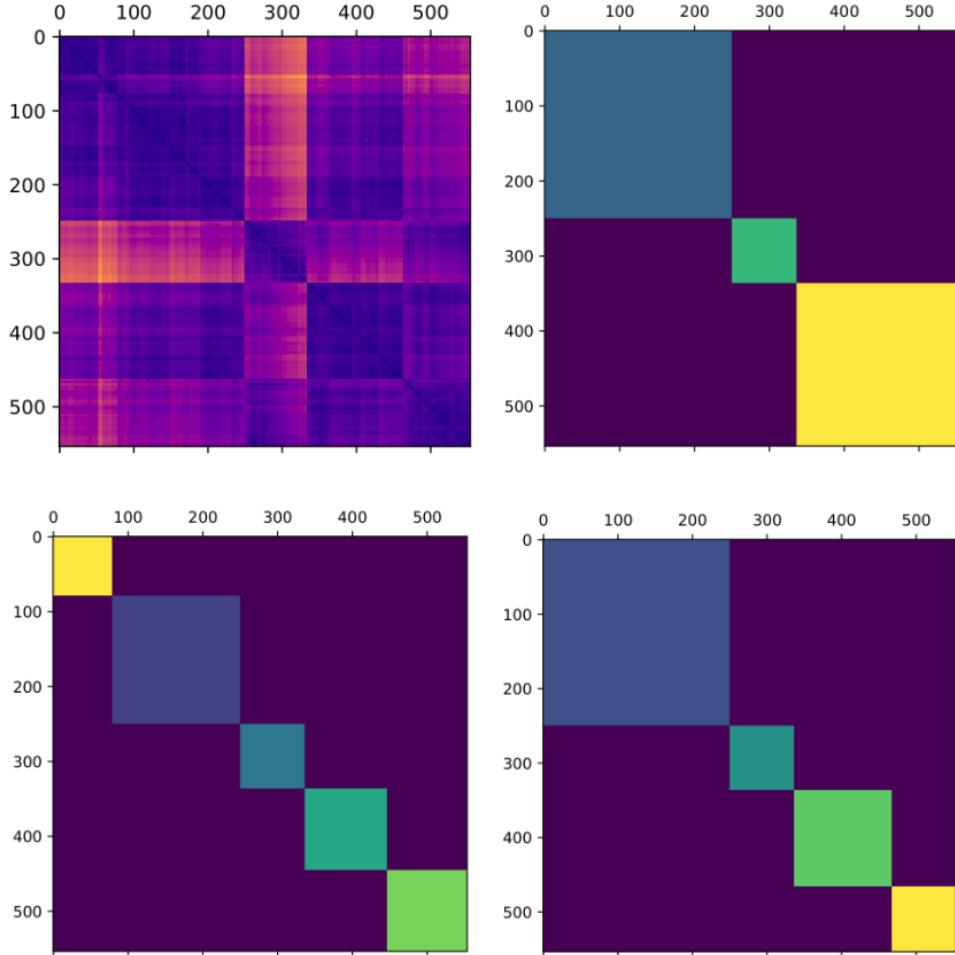


FIGURE 4.8: Clusterization for different numbers of clusters as given by hierarchical clustering

will trying to reconnect other vessels) would need improvements before more analysis is performed on graphs

More features could be extracted from graphs, for example an analysis of the cycles could be very interesting. Moreover, our computation of the curvature and torsion was very sensitive to noise, and one could try to develop a more robust computation (by fitting regular curves on the vessels for example).

Last but not least, there is a lot of arbitrary choices in the way we compute the distance. The Wasserstein distance is computed based on the euclidean metric on the domain of distributions, where each feature is expressed in arbitrary unit. One may want to think about using a non euclidean metric on the domain, to weight the relative importance of features. For example, we may want to penalize less the transformation of a medium radius artery into a big radius artery than the transformation of a medium radius artery into a capillary (small radius). Note that to define such a distance, it is necessary to rely on biologically grounded assumptions.

4.4.2 Long term

The way we compute the Wasserstein distance should be improved. This would allow to take a higher dimensional feature representation as input, and thus, use more features for the graph.

Moreover, a very interesting development would be to develop techniques to extract features in an unsupervised manner, in order to detect patterns which may not be detected by a human observer. For this purpose, tools for inductive learning on graphs have been developed recently [7].

Chapter 5

Vessel binarization

Together with Sophie SKRIABINE, we tried to improve the binarization of raw images in the *ClearVessel* pipeline. The code is available online at https://github.com/skriabineSop/vessel_unsup_seg.

5.1 Deep Learning

We used deep learning on 3D images and more precisely a Wnet-like autoencoder in a similar way as in [8].

5.1.1 Origin and understanding of Deep Learning

In this section we will discuss the purpose of using deep learning. This discussion was inspired by Stéphane Mallat’s attempts to precisely understand convolutional networks [9, 10]. For the sake of simplicity we will restrict ourselves to a classification problem of an instance x with label y .

One of the key points when tackling a classification problem is to reduce the dimensionality of our data. Dimensionality reduction is very powerful because our examples will be much more dense in low dimensional spaces, and thus it will be much easier to find relevant patterns. Note that in high dimensional spaces, volume is very *big* and that even a large amount of data will be very sparse.

To reduce the dimensionality, we find invariances of the distribution $P(Y|X)$ (*i.e.* domains along which the distribution does not depend on X). In fact, those invariances and the hidden relations between examples and labels are two sides of a same coin.

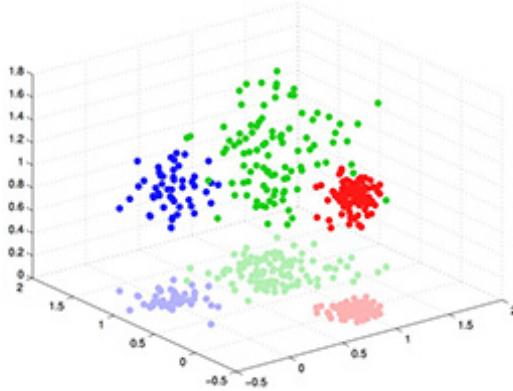


FIGURE 5.1: Projection along the invariance of the probability distribution, thus destroying the meaningless noise in the data

Ideally, we would like those invariances to be linear, so that we can project linearly into a subspace of lower dimension without losing information. Once we are in a small enough space, we hope that we can infer $P(Y|X)$ according to some very simple rule (*e.g.* the value of some coordinate of the input x). In this case, we say that there is *linear separability*.

The problem is that invariances are hardly never linear, so the main goal will be to linearize them. We will briefly explain the strategy of kernel classifiers and then discuss convolutional networks.

5.1.1.1 Kernel classifiers

A way to linearize the invariances of $P(Y|X)$, is to find some change of variable ψ which maps the inputs x to some space of dimension d' , potentially much larger than the initial dimension d of x . We can then operate some low-dimensional linear projection along invariances.

Formally, we say that the change of variable ψ *separates linearly* the function $f : x \mapsto y$ if f is well approximated by a one dimensional projection

$$\exists \omega \in \mathbb{R}^{d'} \text{ s.t. } \tilde{f}(x) = \langle \psi(x), \omega \rangle \quad (5.1)$$

There has been a lot of efforts put on those methods in the last decade, but it turned out to be very difficult to find such a convenient ψ [9].

5.1.1.2 Convolutional networks

Convolutional networks were first introduced by Le Cun [11] in 1990. They have been rediscovered recently thanks to the improvements of computational power and processing speed.

Those networks are optimized in order to find a change of variable ψ by combining in an iterative manner elementary bricks : convolution + non-linearity. More precisely, the network alternates linear operators and contractive non-linearities in successive *layers*. The architectural constraint is imposed, and optimization is done on convolution weights. The intuition is that the network kills variability of the data and extracts invariants hierarchically along the layers. In the same time, the change of variable ψ tends to linearize the function f (f maps the input to the label, and is at first highly non-linear).

Along this process, pooling operations (*i.e.* functions which reduce the dimensionality of the signal, such as downsampling) are also used to progressively shrink the dimension of our space.

Once the change of variable is achieved, we can easily separate examples to classify. The actual process by which such a linearization is achieved remains obscure, and its understanding is an open field of research.

5.1.2 Autoencoders

Autoencoders are a specific type of neural networks where the network is trained to reconstruct its input (*i.e.* X is its own label) while signal being forced to go through a latent space. This latent space is usually of lower dimensionality than the input in order to extract only relevant patterns in the original input [12]. One can also add specific priors on the latent space to give latent representations the desired properties.

5.2 Binarization

Together with Sophie SKRIABINE, we implemented autoencoders for 3D images using *pytorch*. The goal here is to get a binary image from the raw image in order to segment the vessels in an unsupervised manner.

We fed small 3D-patches ($40 \times 40 \times 40$) to the autoencoder (dataset of about $2e5$ patches), which is trained to reconstruct the input patch. The latent representation is binarized

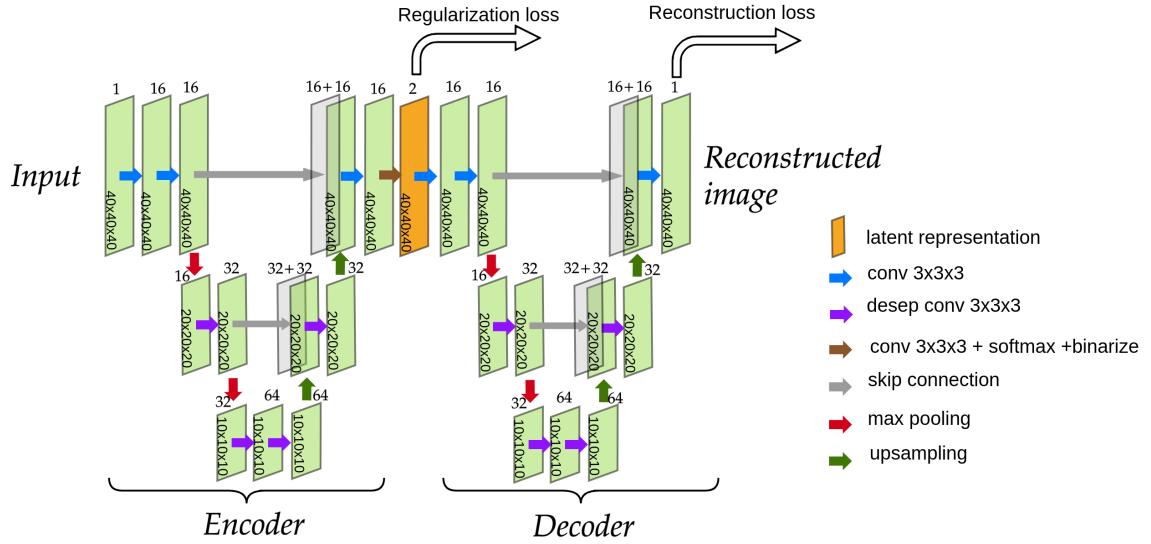


FIGURE 5.2: Architecture of our W-network. It is composed of an Encoder and a Decoder. The latent representation in orange is the *binarization* of the input image. Each green parallelogram represents a 4D tensor. The dimensions of corresponding images are indicated inside the parallelogram (*e.g.* $20 \times 20 \times 20$) and the number of channels is indicated above. *desep conv* stands for depthwise separable convolutions

before being fed to the decoder. Note that the pipeline is still differentiable for backpropagation as we manually defined gradients for the binarization step. The idea is to obtain a binarization which contains a maximum amount of information for reconstruction.

The decoder is only used to optimize the encoder. Once in *production* mode, only the encoder will be used to get the binarization.

5.2.1 Architecture

We adapted a paper for 2D image segmentation using W-net [8]. We used 3D-convolutions instead, and simplified the network as the task is quite *simple* compared to the one of the article (segmentation in natural images).

Our encoder and decoder architectures are inspired by the U-net architecture [13]. Information follows first a contracting path to capture context and then a symmetric expanding path that enables precise localization thanks to skip connections. A *Skip connection* refers to the fact of storing the activation tensor at one point to concatenate it with another tensor deeper in the network.

A detailed presentation of the architecture of our W-network is presented in Figure 5.2. We used basic convolutions (with $3 \times 3 \times 3$ kernels) as well as *depthwise separable* convolutions (a depthwise convolution followed by a pointwise convolution, in order to decouple cross-channel computations and spatial computations). Note that *depthwise separable* convolutions have the advantage of reducing the number of parameters compared to

usual convolutions, which is especially suitable for 3D-images. More precisely, a $3 \times 3 \times 3$ convolution with k filters has $27 \times k \times n_{channel}$ parameters whereas the corresponding *depthwise separable* convolution only has $(27 + k) \times n_{channel}$ parameters, where $n_{channel}$ is the number of channels of the input tensor. Finally, the *binarize* operation refers to applying $x \mapsto \mathbb{1}_{x>0.5}$ elementwise.

To compress spatial information, we used *max pooling* with a kernel size of 2. To expand, we used *upsampling* with trilinear interpolation.

5.2.2 Optimization

5.2.2.1 Optimization parameters

We optimized with *Adam* algorithm [14] with the following parameters : *learning rate*= $1e-3$; $\beta_1 = 0.9$; $\beta_2 = 0.999$ and $\epsilon = 1e-8$.

For the reconstruction loss, we chose a simple *Mean Square Error* between the voxels of the reconstructed image and the ones of the input image.

To monitor the optimization process, we used our visualization tools for 3D image visualization as well as *Tensorboard*.

5.2.2.2 Regularization

In order to obtain a latent representation with the desired properties (*e.g.* continuity of vessels) we tried to put a prior distribution on the latent space. In the optimization process, this is equivalent to adding a regularization term.

We tried several regularizations. As suggested in the article [8], we used *Soft Normalized Cut*, which tends to produce clusters with minimal frontier between them. For details about *Soft Normalized Cut*, please refer to [15]. We chose $\lambda_{SC} = 5e-11$ as the regularization parameter.

As an attempt to reduce the noise in our segmentations, we also used a *Total Variation* regularization [16], which is widely used for image denoising. We chose $\lambda_{TV} = 5e-8$ as the regularization parameter.

A visualization of the binarization with this pipeline is provided [Figure 5.3](#)

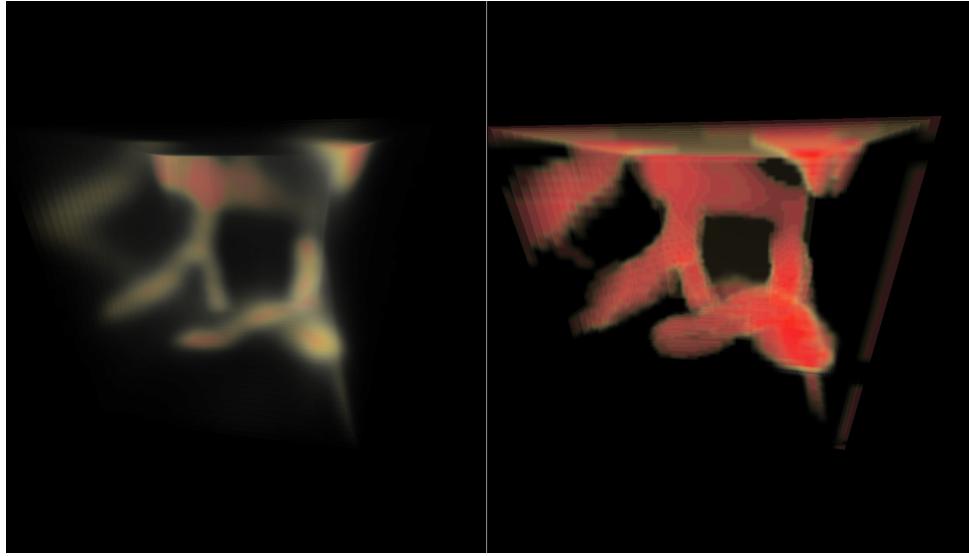


FIGURE 5.3: *Left* : input image. *Right* : binarization obtained with the encoder. We observe that vessels appear larger than in the original image and that a dark section of a vessel is successfully reconstructed (backmost voxels)

5.2.3 Possible extensions

Even with those regularizations, we did not succeed in segmenting vessels without any interruption. To improve this, one would probably need a prior distribution which is structure-aware, meaning that the information of the vessel structure as continuous tubes would be *contained* in the prior distribution.

We should also be concerned with the scalability of this method to full brain images. As the previous technique also used some convolutions, the computation times should be of the same order of magnitude (but we use more convolutions, 7 in total for the encoder, but only 3 convolutions on full-size images, which constitute the computational bottleneck).

Conclusion

During those four months, I had the chance to be part of a pluridisciplinary environment and to discover academic research in both biology and mathematics. I developed visualization and mathematical tools which, I hope, will help biologists analyze the huge amount of brain volume images they work with. On top of that, I enjoyed documenting on various methods, from shape analysis to optimal transport.

The main contributions are the implementation of varifold and normal cycle distances for protocol comparison, and the comparison of the organization of the vasculature in different regions of the brain based on a feature-based Wasserstein distance. I also performed some statistical analysis on the graphs, developed a method for graph reduction, and used autoencoders to try to improve the vessel segmentation.

There is still a lot of work to do on this project. In my opinion, the pluridisciplinary aspect would require several people with different backgrounds to work on it. Most importantly, I think a reflexion about how to get people with different backgrounds to work efficiently together is necessary. Even if I do not have *the* solution, my intuition is that one of the key aspects of a good strategy would be to define clear long-term and middle-term objectives. This way, each one would be able to use their skills and knowledge to achieve their goal.

To summarize, I think this internship was a very good experience which helped me define what I want to do in the future. I would be eager to tackle theoretical issues (defined in cooperation with biologists or clinicians) which would help in a broad range of applications.

Appendix A

List of Brain regions by cluster

Cluster 1

Parastrial nucleus , Paragigantocellular reticular nucleus, dorsal part , Supragenual nucleus , Edinger-Westphal nucleus , Perireunensis nucleus , Infralimbic area, layer 6a , Nucleus raphe pontis , Subparafascicular nucleus, magnocellular part , Retrosplenial area, lateral agranular part, layer 6a , Anteromedial visual area, layer 2/3 , Infralimbic area, layer 6b , Subgeniculate nucleus , Nucleus x , Superior olivary complex, medial part , Nucleus of the trapezoid body , Supratrigeminal nucleus , Motor nucleus of trigeminal , Anteromedial visual area, layer 6a , Primary somatosensory area, upper limb, layer 5 , Medullary reticular nucleus, dorsal part , Tegmental reticular nucleus , Dorsal auditory area, layer 4 , Primary somatosensory area, trunk, layer 4 , Primary somatosensory area, lower limb, layer 4 , Primary auditory area, layer 5 , Primary visual area, layer 4 , Supramammillary nucleus , Superior vestibular nucleus , Anterior cingulate area, ventral part, 6b , Linear nucleus of the medulla , Ventral posterolateral nucleus of the thalamus, parvicellular part , Preparasubthalamic nucleus , Subceruleus nucleus , Lateral visual area, layer 4 , Posterolateral visual area, layer 5 , Primary visual area, layer 5 , Retrosplenial area, lateral agranular part, layer 5 , Dorsal premammillary nucleus , Primary auditory area, layer 6a , Secondary motor area, layer 5 , Orbital area, ventrolateral part, layer 5 , posteromedial visual area, layer 4 , Posterolateral visual area, layer 4 , Gigantocellular reticular nucleus , Pedunculopontine nucleus , solitary tract , Intermediate reticular nucleus , medial longitudinal fascicle , Nucleus raphe magnus , Hypoglossal nucleus , Primary auditory area, layer 4 , Medullary reticular nucleus, ventral part , Accessory supraoptic group , Subparafascicular area , optic nerve , Accessory facial motor nucleus , Infracerebellar nucleus , Trochlear nucleus , doral tegmental decussation , Oculomotor nucleus , Ventral tegmental nucleus , Anteromedial visual area, layer 6b , Lateral visual area, layer 6a , Anterolateral visual area, layer 6b , Posterior auditory area, layer 6b , Posterolateral visual area, layer 6b , Lateral vestibular nucleus , Abducens nucleus , principal mammillary tract , Lateral visual area, layer 6b , Posterolateral visual area, layer 6a , Lateral visual area, layer 5 , Anteromedial visual area, layer 4 , Posterior auditory area, layer 4 , Posterior auditory area, layer 5 , Posterior auditory area, layer 6a , Anteromedial visual area, layer 5 , Central linear nucleus raphe , Paraventricular hypothalamic nucleus, descending division

Cluster 2

ventral tegmental decussation , Anterior pretectal nucleus , Ventral posteromedial nucleus of the thalamus, parvicellular part , mammillothalamic tract , Nucleus sagulum , Dorsal tegmental nucleus , Gustatory areas, layer 6a , Primary somatosensory area, upper limb, layer 4 , Anterodorsal preoptic nucleus , Retrosplenial area, ventral part, layer 6b , Retrosplenial area, lateral agranular part, layer 6b , Nucleus of Darkschewitsch , Central lateral nucleus of the thalamus , Locus ceruleus , Anterior tegmental nucleus , Posterior pretectal nucleus , inferior colliculus commissure , Primary somatosensory area, trunk, layer 6a , tectospinal pathway , Primary somatosensory area, mouth, layer 6b , Cuneiform nucleus , Primary somatosensory area, upper limb, layer 2/3 , Globus pallidus, external segment , Prelimbic area, layer 6a , Field CA2 , Agranular insular area, ventral part, layer 6b , Orbital area, medial part, layer 6a , Fields of Forel , Primary somatosensory area, trunk, layer 5 , Fastigial nucleus , Retrosplenial area, ventral part, layer 5 , Primary somatosensory area, mouth, layer 6a , Midbrain reticular nucleus , retrorubral area , Pontine reticular nucleus , Pontine reticular nucleus, caudal part , Nucleus ambiguus, dorsal division , Medial preoptic nucleus , Superior central nucleus raphe , superior cerebellar peduncles , Nucleus prepositus , Paraventricular hypothalamic nucleus , Subparafascicular nucleus, parvicellular part , Visceral area, layer 6a , Temporal association areas, layer 5 , Temporal association areas, layer 6a , Ventromedial hypothalamic nucleus , Supplemental somatosensory area, layer 6a , Orbital area, lateral part, layer 5 , Anterolateral visual area, layer 6a , Superior colliculus, motor related, deep gray layer , Anterior cingulate area, dorsal part, layer 5 , Superior colliculus, motor related, intermediate white layer , Ventral posteromedial nucleus of the thalamus , Endopiriform nucleus, ventral part , Anterior cingulate area, dorsal part, layer 6a , Primary somatosensory area, barrel field, layer 6a , Primary somatosensory area, upper limb, layer 6a , Endopiriform nucleus, dorsal part , Midbrain reticular nucleus , Paraventricular nucleus of the thalamus , Intermediiodorsal nucleus of the thalamus , dorsal hippocampal commissure , Parataenial nucleus , Primary somatosensory area, barrel field, layer 6b , Dorsal nucleus raphe , Bed nuclei of the stria terminalis , Primary visual area, layer 6b , Dentate gyrus, polymorph layer , Primary somatosensory area, trunk, layer 2/3 , Primary somatosensory area, lower limb, layer 2/3 , Primary visual area, layer 6a , Medial septal nucleus , Basolateral amygdalar nucleus, ventral part , anterior commissure, olfactory limb , Nucleus of Roller , Primary auditory area, layer 6b , Infralimbic area, layer 5 , Visceral area, layer 6b , Periventricular hypothalamic nucleus, anterior part , Ventral auditory area, layer 5 , Orbital area, medial part, layer 5 , Retrosplenial area, dorsal part, layer 5 , Retrosplenial area, ventral part, layer 6a , posteromedial visual area, layer 5 , Primary motor area, Layer 5 , Red nucleus , Parvicellular reticular nucleus , Agranular insular area, dorsal part, layer 5 , Agranular insular area, dorsal part, layer 6a , Orbital area, lateral part, layer 2/3 , Retrosplenial area, dorsal part, layer 6a , Secondary motor area, layer 6a , Septohippocampal nucleus , Interanterodorsal nucleus of the thalamus , Tuberomammillary nucleus, dorsal part , Anterolateral visual area, layer 4 , Dorsal auditory area,

layer 5 , Subthalamic nucleus , Nucleus of the brachium of the inferior colliculus , Primary somatosensory area, lower limb, layer 5 , Principal sensory nucleus of the trigeminal , Spinal nucleus of the trigeminal, caudal part , Superior olfactory complex, lateral part , Spinal nucleus of the trigeminal, oral part , Interposed nucleus , Nucleus of the lateral olfactory tract, layer 3 , Superior olfactory complex, periolivary region , Nucleus raphe obscurus , central canal, spinal cord/medulla , vomeronasal nerve , Posterior auditory area, layer 2/3 , Subparaventricular zone , Accessory olfactory bulb, granular layer , Ventral auditory area, layer 4 , Anterior cingulate area, ventral part, 6a , Posterior amygdalar nucleus , Posteriorlateral visual area, layer 2/3 , Claustrum , Primary somatosensory area, nose, layer 6a , Primary somatosensory area, barrel field, layer 2/3 , Basomedial amygdalar nucleus , Agranular insular area, ventral part, layer 6a , Basolateral amygdalar nucleus, posterior part , Central amygdalar nucleus , Basolateral amygdalar nucleus, anterior part , Superior colliculus, motor related, intermediate gray layer, sublayer b , Agranular insular area, posterior part, layer 5 , Dorsal auditory area, layer 6a , Magnocellular reticular nucleus , Magnocellular nucleus , Paraventricular nucleus , Primary somatosensory area, nose, layer 2/3 , Primary somatosensory area, mouth, layer 5 , Primary somatosensory area, nose, layer 4 , medial lemniscus , arbor vitae , Spinal nucleus of the trigeminal, interpolar part , Gustatory areas, layer 5 , Supplemental somatosensory area, layer 4 , Primary somatosensory area, nose, layer 5 , Primary somatosensory area, barrel field, layer 5 , Supplemental somatosensory area, layer 5 , Primary somatosensory area, barrel field, layer 4 , Parabrachial nucleus , Main olfactory bulb , Nucleus accumbens , Substantia innominata , Agranular insular area, ventral part, layer 5 , Primary motor area, Layer 6a , Prelimbic area, layer 5 , Nucleus y , Dentate nucleus , Anterolateral visual area, layer 5 , internal medullary lamina of the thalamus , Anterior cingulate area, ventral part, layer 5 , Agranular insular area, dorsal part, layer 6b , Prelimbic area, layer 6b , Peripeduncular nucleus , vestibular nerve , Laterodorsal tegmental nucleus , Sublaterodorsal nucleus , Interanteromedial nucleus of the thalamus , Posteroventral preoptic nucleus , Nucleus incertus , Infralimbic area, layer 2/3 , Primary somatosensory area, trunk, layer 6b , Anterior cingulate area, dorsal part, layer 6b , Gustatory areas, layer 6b , Interstitial nucleus of Cajal , supramammillary decussation

Cluster 3

Anteromedial visual area, layer 1 , Primary somatosensory area, lower limb, layer 1 , Retrosplenial area, lateral agranular part, layer 1 , Posteriorlateral visual area, layer 1 , Anterolateral visual area, layer 1 , Lateral visual area, layer 1 , Primary visual area, layer 1 , posteromedial visual area, layer 1 , Lateral dorsal nucleus of thalamus , Dorsal part of the lateral geniculate complex , Superior colliculus, superficial gray layer , fiber tracts , Piriform-amygadalar area , Posterior auditory area, layer 1 , Ventral part of the lateral geniculate complex , Anteroventral preoptic nucleus , Postpiriform transition area , Septofimbrial nucleus , Nucleus of the lateral olfactory tract, molecular layer , Agranular insular area, ventral part, layer 1 , Periventricular hypothalamic nucleus, posterior part , Primary somatosensory area, nose, layer 1 , Supraoptic nucleus , lateral olfactory tract, body , cerebral peduncle , Olfactory areas , Olfactory tubercle , Interfascicular nucleus raphe , fourth ventricle , Lingula (I) , Orbital area, ventrolateral part, layer 1 , optic chiasm , Subfornical organ , Rostral linear nucleus raphe , posterior commissure , External cuneate nucleus , middle cerebellar peduncle , Tuberomammillary nucleus, ventral part , fasciculus retroflexus , Orbital area, medial part, layer 1 , Anterior cingulate area, dorsal part, layer 1 , Retrosplenial area, ventral part, layer 1 , optic tract , Hippocampal formation , Dentate gyrus, molecular layer , gracile fascicle , motor root of the trigeminal nerve , Vascular organ of the lamina terminalis , ventricular systems , Fasciola cinerea , Superior colliculus, motor related, intermediate gray layer, sublayer a , Prelimbic area, layer 1 , Frontal pole, layer 1 , Perirhinal area, layer 1 , Primary auditory area, layer 1 , Agranular insular area, posterior part, layer 1 , brachium of the inferior colliculus , Dorsal auditory area, layer 1 , lateral recess , Primary somatosensory area, mouth, layer 1 , Supplemental somatosensory area, layer 1 , Medial pretectal area , Parapyrimal nucleus , Primary somatosensory area, upper limb, layer 1 , stria terminalis , Temporal association areas, layer 1 , Ectorhinal area/Layer 1 , lateral ventricle , Primary motor area, Layer 1 , Ventral auditory area, layer 1 , Secondary motor area, layer 1 , habenular commissure , Precommissural nucleus , trochlear nerve , ventral hippocampal commissure , Gustatory areas, layer 1 , Agranular insular area, dorsal part, layer 1 , Retrosplenial area, dorsal part, layer 1 , fimbria , Visceral area, layer 1 , Olivary pretectal nucleus , pyramid , Ventrolateral preoptic nucleus , Orbital area, ventrolateral part, layer 6b , Nucleus of the posterior commissure , Temporal association areas, layer 6b

Cluster 4

Orbital area, ventrolateral part, layer 6a , Globus pallidus, internal segment , corpus callosum , cingulum bundle , Periaqueductal gray , Agranular insular area, posterior part, layer 2/3 , Diagonal band nucleus , Frontal pole, layer 2/3 , external medullary lamina of the thalamus , Superior colliculus, motor related, deep white layer , Orbital area, medial part, layer 2/3 , Orbital area, lateral part, layer 6a , Paracentral nucleus , Perirhinal area, layer 6a , Primary motor area, Layer 6b , Central lobule , Median preoptic nucleus , Orbital area, ventrolateral part, layer 2/3 , Orbital area, lateral part, layer 1 , Dorsal peduncular area , Orbital area, lateral part, layer 6b , anterior commissure, temporal limb , Submedial nucleus of the thalamus , Accessory olfactory bulb, mitral layer , Accessory olfactory bulb, glomerular layer , Secondary motor area, layer 6b , Anterior cingulate area, dorsal part, layer 2/3 , Nucleus of the lateral lemniscus , Folium-tuber vermis (VII) , Pontine gray , Ventral premammillary nucleus , corticospinal tract , Paramedian lobule , Facial motor nucleus , Anteroventral periventricular nucleus , Nucleus ambiguus, ventral division , Dorsal motor nucleus of the vagus nerve , Uvula (IX) , Declive (VI) , Substantia nigra, compact part , Nucleus of the solitary tract , Flocculus , Gustatory areas, layer 4 , Primary somatosensory area, mouth, layer 4 , Ventral posterolateral nucleus of the thalamus , Ventral medial nucleus of the thalamus , columns of the fornix , Posterior complex of the thalamus , Prelimbic area, layer 2/3 , Anterior hypothalamic nucleus , Zona incerta , Substantia nigra, reticular part , Copula pyramidis , Ansiform lobule , Pallidum , Visceral area, layer 4 , Primary auditory area, layer 2/3 , Supplemental somatosensory area, layer 2/3 , Anterior olfactory nucleus , Nucleus of the optic tract , Primary somatosensory area, nose, layer 6b , Parabigeminal nucleus , Medulla , Pons , Ventral cochlear nucleus , Medial mammillary nucleus , Anterolateral visual area, layer 2/3 , Simple lobule , Anterior amygdalar area , Spinal vestibular nucleus , Ectorhinal area/Layer 6b , Central medial nucleus of the thalamus , Midbrain trigeminal nucleus , Primary somatosensory area, lower limb, layer 6a , Pontine central gray , Retrosplenial area, dorsal part, layer 6b , Lateral visual area, layer 2/3 , Lateral septal nucleus, caudal (caudodorsal) part , Supplemental somatosensory area, layer 6b , Intercalated amygdalar nucleus , Field CA1 , Medial preoptic area , Superior colliculus, motor related, intermediate gray layer , Visceral area, layer 5 , internal capsule , Caudoputamen , Subiculum , Dorsomedial nucleus of the hypothalamus , Anteromedial nucleus , Suprachiasmatic nucleus , Culmen , Periventricular hypothalamic nucleus, preoptic part , Basic cell groups and regions , Superior colliculus, optic layer , Lateral septal nucleus, ventral part , amygdalar capsule , Ectorhinal area/Layer 6a , Posterior limiting nucleus of the thalamus , Ectorhinal area/Layer 5 , Field CA3 , Primary motor area, Layer 2/3 , Agranular insular area, posterior part, layer 6a , posteromedial visual area, layer 6a , Bed nucleus of the accessory olfactory tract , Periventricular hypothalamic nucleus, intermediate part , Primary visual area, layer 2/3 , Retrosplenial area, lateral agranular part, layer 2/3 , Ventral auditory area, layer 6a , Mediodorsal nucleus of thalamus , midbrain tract of the trigeminal nerve

Cluster 5

Temporal association areas, layer 4 , Fundus of striatum , Rostrolateral area, layer 6b , alveus , Lateral septal nucleus, rostral (rostroventral) part , Perirhinal area, layer 5 , Anteroventral nucleus of thalamus , Dorsal auditory area, layer 2/3 , Secondary motor area, layer 2/3 , Agranular insular area, ventral part, layer 2/3 , Ventral anterior-lateral complex of the thalamus , Lateral hypothalamic area , Posterior hypothalamic nucleus , Cortical subplate , posteromedial visual area, layer 2/3 , Lateral preoptic area , Nucleus of reuniens , Gracile nucleus , Lateral terminal nucleus of the accessory optic tract , Bed nucleus of the anterior commissure , Primary somatosensory area, lower limb, layer 6b , Barrington's nucleus , Induseum griseum , Anterior cingulate area, ventral part, layer 1 , Infralimbic area, layer 1 , Superior colliculus , motor related, intermediate gray layer, sublayer c , Inferior salivatory nucleus , Intergeniculate leaflet of the lateral geniculate complex , Anterior cingulate area, ventral part, layer 2/3 , Medial habenula , Primary somatosensory area, upper limb, layer 6b , Perirhinal area, layer 6b , Lateral habenula , Ectorhinal area/Layer 2/3 , Perirhinal area, layer 2/3 , Parafascicular nucleus , Dentate gyrus, granule cell layer , Presubiculum , Thalamus , Taenia tecta , Entorhinal area, lateral part , Postsubiculum , dorsal fornix , posteromedial visual area, layer 6b , Suprageniculate nucleus , Lateral posterior nucleus of the thalamus , Primary somatosensory area, trunk, layer 1 , Anterodorsal nucleus , stria medullaris , Primary somatosensory area, barrel field, layer 1 , Ventral auditory area, layer 2/3 , Temporal association areas, layer 2/3 , Parasolitary nucleus , Nodulus (X) , Dorsal cochlear nucleus , Inferior colliculus , Medial vestibular nucleus , mammillary peduncle , Gustatory areas, layer 2/3 , Primary somatosensory area, mouth, layer 2/3 , Nucleus raphe pallidus , Ventral tegmental area , Agranular insular area, dorsal part, layer 2/3 , Paragigantocellular reticular nucleus, lateral part , Cuneate nucleus , Hypothalamus , spinal tract of the trigeminal nerve , Paraflocculus , Triangular nucleus of septum , Area postrema , rubrospinal tract , Midbrain , olfactory nerve layer of main olfactory bulb , lateral lemniscus , Arcuate hypothalamic nucleus , Inferior olive complex , Retrochiasmatic area , facial nerve , Interpeduncular nucleus , trapezoid body , inferior cerebellar peduncle , Lateral mammillary nucleus , Dorsal auditory area, layer 6b , Rhomboid nucleus , cerebral aqueduct , Agranular insular area, posterior part, layer 6b , Ventral auditory area, layer 6b , Entorhinal area, medial part, ventral zone , Medial amygdalar nucleus , Entorhinal area, medial part, dorsal zone , Piriform area , Striatum , cuneate fascicle , Retrosplenial area, dorsal part, layer 2/3 , Medial geniculate complex , Lateral reticular nucleus , Pyramus (VIII) , Parasubiculum , Cortical amygdalar area, anterior part , Visceral area, layer 2/3 , Reticular nucleus of the thalamus , Tuberal nucleus , Cortical amygdalar area, posterior part , Retrosplenial area, ventral part, layer 2/3 , third ventricle , Lateral amygdalar nucleus , Striatum-like amygdalar nuclei , mammillotegmental tract

Bibliography

- [1] F. Schmid, M. Barret, P. Jenny, and B. Weber, “Vascular density and distribution in neocortex,” *Neuroimage*, June 2017.
- [2] I. Kaltenmark, B. Charlier, and N. Charon, “A general framework for curve and surface comparison and registration with oriented varifolds,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [3] P. Roussillon, *Modèles de Cycles Normaux pour l’Analyse des Déformations*. PhD thesis, Université Paris Descartes, November 2017.
- [4] C. Villani, *Optimal transport: old and new*, vol. 338. Springer Science & Business Media, 2008.
- [5] J. Solomon, F. de Goes, G. Peyré, M. Cuturi, A. Butscher, A. Nguyen, T. Du, and L. Guibas, “Convolutional wasserstein distances: Efficient optimal transportation on geometric domains,” *ACM Trans. Graph.*, vol. 34, pp. 66:1–66:11, July 2015.
- [6] L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [7] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in *Advances in Neural Information Processing Systems*, pp. 1024–1034, 2017.
- [8] X. Xia and B. Kulis, “W-net: A deep model for fully unsupervised image segmentation,” 2017. arXiv :1711.08506.
- [9] S. Mallat, “Mystères mathématiques des réseaux de neurones convolutionnels,” 2016. <http://www.college-de-france.fr/site/yann-lecun/seminar-2016-02-19-15h30.html>.
- [10] S. Mallat, “Understanding deep convolutional networks,” 2016. arXiv :1601.04920v1.
- [11] Y. L. Cun, B. Boser, J. S. Denker, R. E. Howard, W. Hubbard, L. D. Jackel, and D. Henderson, “Advances in neural information processing systems 2,” ch. Handwritten Digit Recognition with a Back-propagation Network, pp. 396–404, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990.
- [12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.

- [13] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.
- [14] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [15] J. Shi and J. Malik, “Normalized cuts and image segmentation,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, pp. 731 – 737, 2000.
- [16] A. Chambolle, V. Caselles, M. Novaga, D. Cremers, and T. Pock, “An introduction to Total Variation for Image Analysis.” working paper or preprint, Nov. 2009.