Sebastiano Ferraris

The Log-composition of Stationary Velocity Fields in Diffeomorphic Image Registration

Universtiy College London
Medical Physiscs and Biomedical Engineering

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Log-Master of Research

July 7, 2015

Supervisor
Tom Vercauteren

Co-Supervisor
Marc Modat

Acknowledgments

Abstract

Image registration is one of the critical tool in medical imaging. It consists in the process of alignment of two or more patients' images with the aim of determining and quantifying the occurring anatomies' correspondences and differences. It is widely used in both academical studies and applications, and it continuously challenges researchers to enhance accuracy, improve reliability and reduce computational time.

The introduction of the *Lie group of diffeomorphisms* from differential geometry to image registration provides an interesting option to model the organs' deformations and to quantify them, thanks to the *log-euclidean framework* for the computation of statistics. This enables the representation of diffeomorphisms embedded in the one-parameter subgroup as *stationary velocity field* (SVF) in the Lie group's tangent space (the *Lie algebra*). In this space statistics can be computed easily and consistently, but one of the challenges remains the numerical computation of the map that transforms SVF in the corresponding diffeomorphisms, the Lie exponential, as well as the numerical computation of its inverse function, the Lie logarithm.

These two transformations allow in particular the computation of the composition of diffeomorphisms in the tangent space, operation called in this thesis *log-composition.* The necessity of finding fast numerical computation of the log-composition arises for example in the *log-demons* and in the *symmetric log-demon* registration algorithm as well as in the fast computation of the Lie logarithm itself.

In this research we analyze existing numerical method for the computation of the log-composition, based on the BCH formula and we compare them with two new methods developed in this research, one based on the Taylor expansion and the other on the geometrical concept of parallel transport.

Contents

1 Introduction to Diffeomorphic Image Registration **1**
1.1 Toward an ill-posed Problem . . . 1
1.1.1 Examples of applications . . . 1
1.1.2 Diffeomorphisms in medical imaging: State of the Art . . . 2
1.1.3 Using Diffeomorphisms: Utility and Liability . . . 3
1.2 Image Registration Framework . . . 4
1.2.1 Iterative Registration Algorithm . . . 5
1.2.2 LDDMM: Classic, Shooting and Stationary . . . 5
1.2.3 Demonology: Classic, Additive, Diffeomorphic, Log and Symmetric . . 8
1.3 The Log-composition at the Service of Image Registration . . . 9

2 Tools from Differential Geometry **11**
2.1 A Lie Group Structure for the Set of Transformation . . . 11
2.2 Lie Exponential, Lie logarithm, Lie log-composition and the BCH formula . 11
2.3 Affine Exponential, Affine Logarithm and Parallel Transport: Definition and Properties . . . 14
2.4 Numerical Computations of the Log-composition . . . 18
2.4.1 Truncated BCH formula for the Log-composition . . . 18
2.4.2 Taylor Expansion Method for the Log-composition . . . 18
2.4.3 Parallel Transport Method for the Log-composition . . . 19

3 Spatial Transformations for the Computations of the Log-composition: SE(2) and SVF **21**
3.1 The Lie Group of Rigid Body Transformations . . . 21
3.1.1 Computations of Log-composition in $\mathfrak{se}(2)$ . . . 23
3.2 One-parameter Subgroup Generated by Stationary Velocity Fields . . . 25
3.2.1 Computations of Log-composition for SVF . . . 26

4 Log-Algorithm using Log-composition **27**
4.1 Spaces of Approximations . . . 27
4.2 The Log-computation Algorithm using Log-composition . . . 28
4.2.1 Truncated BCH Strategy . . . 29
4.2.2 Parallel Transport Strategy . . . 29
4.2.3 Symmetrization Strategy . . . 29

5 Experimental Results **31**
5.1 Log-composition for $\mathfrak{se}(2)$ . . . 32
5.1.1 Methods . . . 32
5.1.2 Results . . . 32
5.1.3 Empirical Evaluations of Computational Time . . . 32
5.2 Log-composition for SVF . . . 32
5.2.1 Methods . . . 33
5.2.2 Results . . . 33

5.2.3 Empirical Evaluations of Computational Time 33
5.3 Log-Algorithm for SVF 33
5.3.1 Methods 33
5.3.2 Results 33
5.3.3 Empirical Evaluations of Computational Time 33
5.4 Conclusion and Further Research 33

Chapter 1

Introduction to Diffeomorphic Image Registration

The series is divergent, therefore we may be able to do something with it.
- Oliver Heaviside

1.1 Toward an ill-posed Problem

The process of determining correspondences between two or more images acquired from patients scans is a challenging task that has seen the application of a growing number of mathematical theories contributing to its solution.
The challenge and the concomitant difficulties in approaching the problem is a consequence of the fact that dealing with image registration problem means dealing with an ill-posed problem. Transformations between anatomies are not unique, and the impossibility to recover spatial or temporal evolution of an anatomical transformation from temporally isolated images, makes any validation a difficult, if not an impossible task. In addition each situation inevitably brings some prior knowledge within the initial data, that may imply some modifications in the problems' setting and may imply some additional constraints. This, of course, impact dramatically the range of possible results.

It is often the practical situation that provides the hint in choosing the optimal constraints, but it almost never provides enough information to reduce the large amount of options involved. A wide range of variants in methodologies and approaches to solve the registration problem has been thus proposed in the last decades: a quick glance to Google scholar reveals about 1200000 papers in *medical image registration* (55% of the whole *image registration* resources).

1.1.1 Examples of applications

One of the main application of image registration is in the domain of brain imaging: there this tool can be used to examine differences between subjects and distinguish their biological features (cross-sectional studies) or to compare different acquisition of the same subject, before and after a surgery or after a fixed period of time (longitudinal studies).

In both cases an accurate comparison between images and the parameters of the transformation involved may result in a quantification of anatomical variability and in a better understanding of the patients' features. For example, brain atrophy is considered a biomarker to diagnose the Alzheimer disease and to analyze its evolution; most of the algorithms and techniques involved in the atrophy measurement requires longitudinal or cross-sectional scans to be aligned, and so are directly affected by the solution of the registration algorithm (See [FF97], [GWRNJ12], [PCL+15] for studies on brain atrophy involving registration).

Also when dealing with motion correction, if a sequence of images is affected by the motion of cardiac pulses or respiratory cycles, registration algorithms are often used for the realignment. They are also involved, for example, in lungs radiotherapy. In this case, a the correspondence between the lungs' deformation obtained from the registration algorithms and the respiratory signal defines a model to direct the X-ray or electrons beam on the cancerous tissue. Thanks to the correspondence, the beam follow the cancerous tissue during the respiration, minimizing damages on the sane tissue [MHSK], [MHM+11].

Another application of image registration is the composition of several images aided to obtain a bigger picture - this procedure is sometime called *mosaicing*. In this case a registration algorithm aligns images using information obtained form the overlapping regions (See [VPM+06], [Sze94] as examples of image registration applied to mosaicing processes).

1.1.2 Diffeomorphisms in medical imaging: State of the Art

In the attempt to classify image registration algorithms, one of the most relevant feature that distinguish them is the choice of the family to whom the transformation belongs and its parametrization. Since anatomies are in a continuous process of modification over time, in general without any variation in the topological features, the use of diffeomorphisms (as invertible topology-preserving functions) appears to be one of the most natural choice to model organs' transformations.

In the development of diffeomorphic image registration, we can broadly identify some steps that led to the concept of log-composition presented in this research:

1981-1996 ▷ The use of diffeomorphisms in medical image registration began from the research of a solution to a class partial differential equations: deformations are modeled as the consequent effect of two balancing forces applied to an elastic body [Bro81] or to conserve the energy momentum [CRM96]. In this early stage, diffeomorphisms are the domain of the solution of a differential equation, and are not considered with their Lie group structure.

1998-2004 ▷ Based on the concept of attraction, the demons algorithm [Thi98], [PCA99] propose the computation of the transformation between images in an iterative framework, where the update of the transformation at each step is parametrized with a vector field that is optimized at each step. This vector field is defined as the set of vectors (demons) that moves each voxel of the moving image into a new position, corresponding to the fixed of the other.
Here diffeomorphisms are not directly involved and the vectors at each voxel are considered as independent elements. In the same year of [Thi98], the set of diffeomorphism was taken into account in image matching and computational anatomy, not only as the set of solutions of some family of differential equations, but with its tangent space [DGM98, Tro98, GM98].

2005-2006 ▷ The almost concomitant publication of the Large Deformation Diffeomorphic Metric Mapping (LDDMM) in [BMTY05] with some investigations about the tangent space to the Lie group of diffeomorphisms as a space where to perform statistics - the so called log-Euclidean framework [ACPA06b, AFPA06] bring to the attention the use of the tangent space to diffeomorphisms and to consider them as a Lie group provided by

a Lie algebra.
The LDDMM revealed all the opportunities provided by differential geometry in considering tangent vectors to the space of transformation in a framework for the computation of image registration. In this setting, the tangent vector field comes from the solution of the ODE that models the transformations and it consists of the set of the non-stationary vector field (also time varying vector field or TVVF). After the log-Euclidean framework [ACPA06b], aimed at the computation of statistics of diffeomorphisms, the set of tangent vector field is restricted to the time-independent vector field (also stationary velocity field or SFV).

2007-2013 ▷ The restriction to SVF was subsequently considered in some further improvements of LDDMM as DARTEL and Stationary LDDMM [Ash07], [HBO07]. Log-Euclidean framework brought new life also to the demons algorithm, that become, in 2007, the diffeomorphic demons [VPPA07]. Subsequent approaches involving the symmetrization of the energy function and the use of a different norm (local correlation coefficient instead of L^2) are proposed in symmetric log-demons [VPPA08] and LCC-demons [LAF$^+$13] respectively.

1.1.3 Using Diffeomorphisms: Utility and Liability

If the algorithm is meant to model transformations that preserves distances, orientations and angles, then the set of transformations involved can be bonded to the rigid body transformations group $SE(3)$. The consequent registration algorithm will be suitable for example to compensate the motion in a rapid sequence of scans, or if some little differences has happened, to compare them in longitudinal and cross sectional scans.

If the algorithm is meant to model transformations that only preserves topology, as often happen for medical images, then the set of transformations can be bonded to the set of diffeomorphism (bijective differentiable maps with differentiable inverse, for example [Lee12]) indicated by *Diff*. This is, on one hand, particularly appealing for its algebraic group structure (see, for example [Art11]) and for its analytical properties. On the other hand, due to its infinite-dimensional nature, a mathematical formalization of *Diff* as Lie group (so as differentiable manifold within a group structure, see [War13]) is not of immediate understanding, and it is still an open field of research.

Attempt to provide some handles to this group for easy manipulation was done for the first time in 1966 by Vladimir Arnold [Arn66] (see also [Arn98], more readable for non-French speakers): to solve differential equation in hydrodynamic, the set *Diff* is considered as a Lie group possessing a Lie algebra. This assumption is not formally prosecuted in accordance to the problem-oriented nature of this paper. Subsequent steps in the exploration of the set of diffeomorphisms as a Lie group can be found in [MA70, EM70, Omo70, Les83]. A survey on early development of infinite dimensional Lie group can be found in [Mil84a], while more recent results and applications on diffeomorphisms has been published in [OKC92, BHM10, Sch10, BBHM11].

Considering *Diff* as a differentiable manifold involves the idea of having it locally in correspondence with some generalized “infinite-dimensional Euclidean” space. Attempt to set this correspondence showed that for some infinite-dimensional group the transition maps are smooth over the more general Banach spaces [KW08]. This led to the idea of Banach Manifolds. Unfortunately the group of diffeomorphisms do not belongs to the category of Banach manifold but requires an even more general space on which the transition maps are smooth: the Frechet space. Here, important theorems from analysis, as the inverse function theorem, the Frobenius theorem, or the main results from the Lie group theory in a finite dimensional settings, as Lie correspondence theorems, do not holds anymore. These difficulties led some researchers in approaching the set of diffeomorphisms from other perspectives:

for example, instead of treating *Diff* as a group equipped with differential structures, it is seen as a quotient of other well behaved group [Woj94]. In other cases, in [MA70] first and in [Mil84b] later, Banach spaces are substituted with more general locally convex spaces to underpin the definition of smooth manifolds.

Without denying the importance of fundamentals and underestimating the doors research in this domain may open, we will approach the matter in as similar way of what has been done in set theory: we will use a *naive approach* to infinite dimensional Lie group, where the fundamental definition of infinite dimensional Lie group is a generalization of the finite dimensional case left more to the intuition than to a robust formalization.

Other than theoretical difficulties, there is another limitation in the utilization of diffeomorphisms, that comes down to the practical necessity of dealing with discrete images and to implement them in softwares. *Two subset of some topological space have the same topology if exists an homeomorphism between them*: this analytical property do not holds if the objects involved are considered in a discretized space. Separated subset remains separated until their distance is less than the size of a voxel for a significant region; if this do not happen, even with a homeomorphic underpinning model, the discretization process do not preserve the topology.

1.2 Image Registration Framework

A *d-dimensional image* is a continuous function from a subset Ω of the coordinate space $\mathbb{R}^d$ (having in mind particular cases $d = 2, 3$) to the set of real numbers $\mathbb{R}$. Given two of them, $F : \Omega_F \to \mathbb{R}$ and $M : \Omega_M \to \mathbb{R}$, called respectively *fixed image* and *moving image*, the *image registration problem* consists in the investigation of features and parameters of the transformation function

$$\begin{aligned} \varphi : \mathbb{R}^d \supseteq \Omega_F &\longrightarrow \Omega_M \subseteq \mathbb{R}^d \\ \mathbf{x} &\longmapsto \varphi(\mathbf{x}) \end{aligned}$$

such that for each point $\mathbf{x} \in \Omega_F$ the element $M(\varphi(\mathbf{x}))$ and $F(\mathbf{x})$ are as closed as possible according to a chosen measure of similarity. Usually the function defined as the composition of the moving image after the transformation, $M \circ \varphi$, is called *warped image*.

The definition of image registration problem proposed here can be represented by the following diagram, where φ is the solution that makes f the identity function:

$$\begin{tikzcd} \Omega_F \arrow[r, "\varphi"] \arrow[d, "F"] & \Omega_M \arrow[d, "M"] \\ \mathbb{R} \arrow[r, dashed, "\mathrm{f}"] & \mathbb{R} \end{tikzcd}$$

If $\Omega_F \neq \Omega_M$, it is always possible to apply an homeomorphism to transform them into a common domain Ω, called *background space*, on which both of the images are defined.

This leaves two key degrees of freedom in searching for a solution: the transformation's domain (also called *deformation model*), and the metric to measure the similarity between images.
Once these are chosen, they can be used as constituent of an *image registration framework*:

an iterative process that provides at each step a new function φ that approaches f to the identity.
Each iteration involves the optimization of an *energy function* (or objective function) that measure the similarity between the fixed image and the warped image computed at the previous step. Moreover the metric can be considered with an additive regularization term, that introduces a constraint defined according to the model:

$$\mathcal{E}(F, M, \varphi) = \mathrm{Sim}(F, M, \varphi) + \mathrm{Reg}(\varphi) \tag{1.1}$$

Sim is a function that measure the similarity and Reg is the regularization term, function of the transformation. The optimization algorithm on which the framework is based and the resampling strategy - process of resize the image from one dimension to another - provide additional degrees of freedom in defining the framework.

1.2.1 Iterative Registration Algorithm

The definition of registration problem and the iterative framework described above, raise several issues. For example there are no reasons to believe that such a correspondence is unique and that there is at least one of them whose behaviour corresponds to a reasonable biological transformation between anatomies.

One way to deal with this problem is to add some constraints on the transformation φ, such that it is bound to models realistic changes that can occur in biological tissues. The kind and quality of the constraints are one of the features that distinguish one registration algorithm from the other.
The image registration framework here presented can be see as a electronic device with 5 knobs, each with its range:

$$\begin{aligned}
\{\varphi\} &\in \{\text{Transformations}\}\\
\mathrm{Sim} &\in \{\text{Similarity measures}\}\\
\mathrm{Reg} &\in \{\text{Regularization Terms}\}\\
\mathrm{Opt} &\in \{\text{Optimization techniques}\}\\
\mathrm{Res} &\in \{\text{Resampling techniques}\}
\end{aligned}$$

Under the hood of this ideal device we may see an engine that can be schematically represented as in figure 1.1.
Modulating on the value of each knob, changing for example the set of transformation or the resampling technique, we change between the possible registration algorithms that generally falls in this framework.

What said so far is far from being a complete overview of all of the possible frameworks; it does not take into account the fact that each version or implementation inevitably involves different needs and consequent challenges. Although it is our starting point to introduce two families of registration algorithms, the LDDMM and the demons. For accurate surveys in medical image registration see for example [SDP13], [ZF03] .

1.2.2 LDDMM: Classic, Shooting and Stationary

As previously done in the elastic registration [Bro81], the LDDMM framework [BMTY05] originates by considering motion between images as the motion of a fluid, and utilizes ODE from fluid dynamics to compute the deformation between fixed and moving. A *time varying*

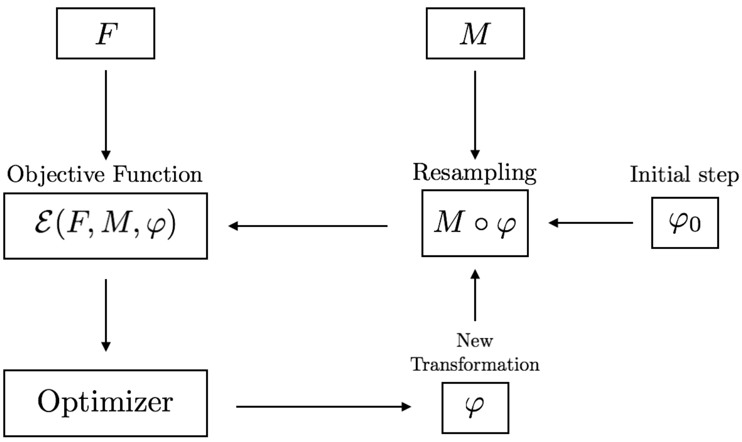

Figure 1.1: Image registration framework scheme.

vector field (TVVF) is the continuously differentiable map defined as

$$\begin{aligned} V : [0,1] &\longrightarrow \mathcal{V}(\Omega) \\ t &\longmapsto V_{(t)} : \Omega \longrightarrow \mathbb{R}^d \\ &\qquad\qquad \mathbf{x} \longmapsto V_{(t,\mathbf{x})} \end{aligned}$$

Once initial conditions are given, at each TVVF, corresponds a unique time varying (or non-stationary) homomorphisms defined by the following ODE

$$\frac{d\phi_t(\mathbf{x})}{dt} = V_{(t,\phi_t(\mathbf{x}))} \tag{1.2}$$

where the function ϕ is a path on the set of homeomorphisms (continuous function from the background space Ω to itself with continuous inverse), indicated with $\mathrm{Hom}(\Omega)$:

$$\begin{aligned} \phi : [0,1] &\longrightarrow \mathrm{Hom}(\Omega) \\ t &\longmapsto \phi_t : \Omega \longrightarrow \Omega \\ &\qquad\qquad \mathbf{x} \longmapsto \phi_t(\mathbf{x}) \end{aligned}$$

The sought transformation φ between fixed and moving images (such that it satisfies $F \circ \varphi^{-1} \simeq M$), is modeled in the LDDMM as the solution of the equation 1.2 at time 1. Using the fundamental theorem of calculus we obtain:

$$\varphi := \phi_1 = \phi_0 + \int_0^1 V_{(t,\phi_t(\mathbf{x}))} dt$$

The set of homeomorphisms is taken into account since it can define a partition of the set of images into equivalence classes, as orbits of the action on the set of images from the background space $\mathcal{I}_\Omega$ (see for example [Art11]). In other words, given a subgroup of homeomorphisms $\mathbb{G} \subseteq \mathrm{Hom}(\Omega)$ and an image F, the orbit of the action of $\mathbb{G}$ over F, given by

$$\mathcal{O}_{\mathbb{G}}(F) = \{F \circ \varphi^{-1} \mid \varphi \in \mathbb{G}\}$$

consists in the set of the images having the same topology of F.

The model proposed in the LDDMM framework imposes a first constraint in considering $\mathbb{G}$ as the set of diffeomorphisms, and a second constraint considering ϕ_t as the shortest path between the identity function on $\mathbb{G}$ and φ. In this way the resulting vector field $V_{(t,\phi_t(\mathbf{x}))}$ is the one that minimize the distance l between transformations:

$$\hat{l} = \inf_{V_{(t,\phi_t(\mathbf{x}))} \,:\, \dot{\phi}_t(\mathbf{x}) = V_{(t,\phi_t(\mathbf{x}))}} \int_0^1 ||V_{(t,\phi_t(\mathbf{x}))}||^2_{L^2} dt$$

For the similarity term the LDDMM uses the L^2 norm (see for example [SS09], chapter 4) between the moving image and the fixed image in the same orbit:

$$\text{Sim}(F, M, \varphi) = \frac{1}{\sigma^2} ||F(\varphi^{-1}) - M||^2_{L^2}$$

while the regularization term is defined on the norm of the velocity vector field tangent to the transformation. Therefore, the optimization function is:

$$\mathcal{E}(F, M, \varphi) = \frac{1}{\sigma^2} ||F(\varphi^{-1}) - M||^2_{L^2} + \int_0^1 ||LV_{(t,\phi_t(\mathbf{x}))}||^2_{L^2} dt$$

As in many other implementation, also in the LDDMM the data structure utilized to store deformation fields are 5-dimensional matrices

$$M = M(x_i, y_j, z_k, t, d) \qquad (i,j,k) \in L, \quad t \in T \quad d = 1,2,3 \tag{1.3}$$

where (x_i, y_j, z_k) are discrete position of a lattice L in the domain of the images, t is the time parameter in a discretized domain T and d is index of the coordinate axis. So, the discretized *tangent vector* $\mathbf{v}_\tau(x_i, y_j, z_k)$ at time t, has coordinates defined by

$$\mathbf{v}_t(x_i, y_j, z_k) = (M(x_i, y_j, z_k, t, 1), M(x_i, y_j, z_k, t, 2), M(x_i, y_j, z_k, t, 3))$$

At the k-th step, the algorithm provides the 5-dim matrix $\mathbf{v}_k$ that is the approximation of the discretized time varying velocity fields $V_{(t,\phi_t(\mathbf{x}))}$. The update at each step is computed as

$$\mathbf{v}_{k+1} = \mathbf{v}_k - \epsilon \vec{\nabla}(\Delta\mathcal{E})$$

where $\Delta\mathcal{E}$ is the discretized version of the energy function and ϵ is the gradient descent step size.

A direct upgrade of the classical LDDMM just introduced performs the optimization on the geodesic flows, defined by a set of Hamiltonian equation (Shooting LDDMM [VRRC12]). In this algorithm the iterative evolution of the deformation field, solution of the optimization algorithm, is regularized with the constraint imposed by an additional scalar field called *initial momentum*. As proved by the authors, the evaluation of this constraint at each step provides geodesics flows of homeomorphisms, but it is computationally expensive. Taking advantages of the log-Euclidean framework presented in [AFPA06], a subsequent algorithm called DARTEL (Diffeomorphic Anatomical Registration using Exponentiated Lie Algebra [Ash07]) uses a constraint based on the stationarity of the involved velocity field. Instead of considering time varying velocity fields constrained by a set of Hamiltonian equations, the domain of vector field is reduced to the stationary velocity fields, whose consequence is a considerable reduction in the computational complexity. A similar algorithm, published in contemporary with DARTEL is [HBO07], based as well on the parametrization of geodesics path of diffeomorphisms with stationary velocity fields.
As happened in the case of the LDDMM, the log-Euclidean framework and the use of SVF and influenced also a second family of registration algorithm, called the *demons algorithm*. In 2007, a new version of the demons, based on diffeomorphisms was proposed in [VPM+06]. Aim of the next section is to introduce the diffeomorphic demons algorithm, second family of algorithm that exploit diffeomorphisms for the computation of the anatomical deformations.

1.2.3 Demonology: Classic, Additive, Diffeomorphic, Log and Symmetric

The first demons-based algorithm in image registration was proposed by [Thi98], in analogy with the Maxwell's demon in thermodynamics. This early version - often called *classic demons* - do not involves diffeomorphisms. The moving image is deformed with a vector field resulting from the computation of the optical flow ([HS81]) regularized by a gaussian filter at each step. The optical flow is based on the idea that a voxel in the moving image is attracted by some force to all the points in the fixed with similar intensity.

The final deformation, solution of the registration problem is obtained composing at each step the previous transformation with an update: let $\{T_k\}_{k=1}^{N}$ be the sequence of deformation and let δT_k be the update at the step k. They can be expressed as the identity plus the displacement field V:

$$T_k(\mathbf{x}) = \mathbf{x} + V_k(\mathbf{x})$$
$$\delta T_k(\mathbf{x}) = \mathbf{x} + \delta V_k(\mathbf{x})$$

And the k-th deformation is computed by composition as:

$$\begin{aligned} T_{k+1}(\mathbf{x}) :&= (\delta T_k \circ T_k)(\mathbf{x}) \\ &= \mathbf{x} + \delta V_k(\mathbf{x}) + V_k(\mathbf{x} + \delta V_k(\mathbf{x})) \end{aligned}$$

Since the third addend is close to $V_k(\mathbf{x})$, some implementation - as for example ITK - consider only the sum between V_{k+1} and V_k in the computation of the update:

$$\begin{aligned} T_{k+1}(\mathbf{x}) :&= (\delta T_k + T_k)(\mathbf{x}) \\ &= \mathbf{x} + V_k(\mathbf{x}) + \delta V_k(\mathbf{x}) \end{aligned}$$

Demons algorithms with this implementation are often called *additive demons*.

In the paper that presents the PASHA demons [CBD+03], authors point out the fact that the classic demon is based on the minimization of a local energy function that sees each voxel independent from each others. In consequence of this limitation the authors of the PASHA reformulate the algorithm using a global energy function, aimed to make the method based on a global energy function, and therefore easier to be analyzed and compared with other methods. With some modifications, the Classic demons algorithm is accompanied back to the framework presented in the previous section, having a global energy function which optimization provides at each step the update of the transformation.

It is important to notice that the PASHA algorithm does not involve any diffeomorphism. The solution is smoothed with the widespread stratagem of applying a Gaussian filter G at each of the global transformation T_k involved in the registration:

$$T_{k+1}(\mathbf{x}) := G_1(T_k(\mathbf{x}) + G_2(\delta T_k(\mathbf{x}))$$

In general if G_1 is omitted the model is sometime called *fluid*, while if G_2 is omitted is called *elastic*.

Diffeomorphisms were introduced within the demons algorithm (*diffeomorphic demons* [VPM+06]) after the presentation of the log-Euclidean framework [AFPA06]. To each stationary velocity field $V \in \mathcal{V}(\Omega)$ is associated a diffeomorphisms φ by the ODE $d\varphi/dt = V_{(t,p)}$. Using Lie theory, SVF are elements of the *Lie algebra* - usually denoted with $\mathfrak{g}$ - while the set of diffeomorphisms are elements of the Lie group - denoted with $\mathbb{G}$ -.

Roughly speaking, the Lie algebra $\mathfrak{g}$ is the tangent space (as local linear approximation) to the Lie group $\mathbb{G}$, and these two spaces are in local correspondence thanks to two "crossing-structure" functions: the *Lie exponential* and the *Lie logarithm. Lie exponential* maps vector fields on the corresponding Lie group elements, while the *Lie logarithm* - inverse of the Lie exponential under some condition, see [DCDC76] or [Lee12] - maps each diffeomorphisms in the correspondent tangent vector field:

$$\varphi = \exp(V) \quad V = \log(\varphi) \qquad\qquad \varphi \in \mathbb{G} \quad V \in \mathfrak{g}$$

In this settings, the update can not be computed simply with a sum of vector fields, since it must reflect the composition of the corresponding diffeomorphisms in the Lie group.

Several approaches has been presented to face the problem of the computation of the update. Diffeomorphic demons compute the transformation at each step of the iterative algorithm as the composition between the diffeomorphism φ_k obtained at the previous step with the exponential of the SVF δV_k, obtained with the optimization algorithm:

$$\varphi_{k+1} := \varphi_k \circ \exp(\delta V_k)$$

In a subsequent version, the log-demons [VPPA08], the composition is performed in the tangent space toward exponential and logarithm functions

$$V_{k+1} := \log(\exp(V_k) \circ \exp(\delta V_k)) \tag{1.4}$$

For this last computation, another treasure from the theory of Lie group has been stolen: the BCH formula. It provides the solution for Z of the equation

$$\exp(Z) = \exp(X) \circ \exp(Y)$$

Its solution involves an infinite series of nested Lie bracket, that do not makes its computation straightforward. To face the problem of its numerical approximation, whose solutions are utilized to solve 1.4, we define in this thesis a binary operation called log-composition.

1.3 The Log-composition at the Service of Image Registration

Every non-rigid registration algorithm requires to be implemented and to work with discretized images. The nature of the computers' memory prevent from the possibility of storing the continuous fluid transformations that, for example, solves the differential equations of the LDDMM or any of the diffeomorphisms resulting from the demons. The only manipulations we are allowed to perform, are storing the discretized vector fields and resampling them using one of the available techniques.
When relying on diffeomorphic algorithms, we still have to consider discretized vector fields as elements of a Lie algebra whose Lie group appears to be a support for the computation of their composition in the equation 1.4. This operation of composition of tangent vector fields, based on the diffeomorphisms is baptized here under the name of *log-composition* and it is defined as

$$X \oplus Y := \log(\exp(X) \circ \exp(Y)) \qquad\qquad \forall\; X, Y \in \mathfrak{g}$$

The main aim of this document is to present a comparison between numerical methods for its computation.

It is worthed to notice that a fast and accurate computation of the log-composition may not influence only the diffeomorphic demons. It may be used as well in medical imaging for

1. Symmetric diffeomorphic demon [VPPA08].
2. Fast computation of the logarithm [BO08] (discussed in chapter 4).
3. Calculus on diffusion tensor [AFPA06].
4. Compute the discrete ladder for Parallel Transport in Transformation Groups [LP14a].

The next chapter is aimed to the formal definition of the log-composition, underpinned with the tools from differential geometry theory and to present two new numerical technique to compute it.

Chapter 2

Tools from Differential Geometry

Give me six hours to chop down a tree
and I will spend the first four sharpening the axe.
-Abraham Lincoln

2.1 A Lie Group Structure for the Set of Transformation

We consider every group $\mathbb{G}$ as a group of transformations acting on $\mathbb{R}^d$, having in mind the particular case $d = 2, 3$ for 2-dimensional or 3-dimensional images. We will focus our attention to transformations defined by matrices or diffeomorphism. Other than group they also have the structure of Lie group: they are considered with a maximal atlas that makes them differentiable manifold, in which the composition of two transformation and the inverse of each transformation are well defined differentiable maps:

$$
\begin{aligned}
\mathbb{G} \times \mathbb{G} &\longrightarrow \mathbb{G} \\
(x, y) &\longmapsto xy^{-1}
\end{aligned}
$$

Differential geometry is in general a technique to use the well known calculus features and operators on spaces different from the usual $\mathbb{R}^n$. Adding the differentiable structure to a group of transformations gives us new handles to hold and manipulate them: in particular provides the opportunity to define a tangent space to each point of the group (and so a fiber bundle), a space of vector fields, a set of flows and one parameter subgroup as well as other features that enrich this structure. Due to space limitations we will refer to [DCDC76], [Lee12] for the definitions and concepts of differential geometry and [dCV92] for definition and concepts of Riemannian geometry.

2.2 Lie Exponential, Lie logarithm, Lie log-composition and the BCH formula

Let $\mathbf{v}$ be an element in the tangent space fo the Lie group $\mathbb{G}$ indicated with $\mathfrak{g}$. The *Lie exponential* is defined as

$$
\begin{aligned}
\exp : \mathfrak{g} &\longrightarrow \mathbb{G} \\
\mathbf{v} &\longmapsto \exp(\mathbf{v}) = \gamma(1)
\end{aligned}
$$

where $\gamma : [0,1] \to \mathbb{G}$ is the unique one-parameter subgroup of $\mathbb{G}$ having $\mathbf{v}$ as its tangent vector at the identity ([dCV92], [EMP06], [AFPA06]). It satisfies the following properties:

1. $\exp(t\mathbf{v}) = \gamma(t)$.
2. $\exp(\mathbf{v}) = e$ if $\mathbf{v} = \mathbf{0}$.
3. $\exp(\mathbf{v}) \circ \exp(-\mathbf{v}) = e$
4. It satisfies the one parameter subgroup property:
$$\exp((t+s)\mathbf{v}) = \gamma(t+s) = \gamma(t) \circ \gamma(s) = \exp(t\mathbf{v})\exp(s\mathbf{v})$$
5. $\exp(\mathbf{v})$ is invertible and $(\exp(\mathbf{v}))^{-1} = \exp(-\mathbf{v})$.
6. $\exp(\mathbf{u}+\mathbf{v}) = \lim_{m\to\infty}(\exp(\frac{\mathbf{v}}{m}) \circ \exp(\frac{\mathbf{v}}{m}))^m$
7. exp is a local isomorphism: which means that it is an isomorphisms between a neighborhood of $\mathbf{0}$ in $\mathfrak{g}$ to a neighborhood of e in $\mathbb{G}$.

The neighborhoods of $\mathbb{G}$ and of $\mathfrak{g}$ such that the last property holds, are called *internal cut locus* of $\mathbb{G}$ and $\mathfrak{g}$ respectively. The *cut locus* is the boundary of the internal cut locus.

When we deal with a matrix Lie group of dimension n, the composition in the Lie group consists in the matrix product and we have the following remarkable properties [Hal15], [KKK08]:

1. for all $\mathbf{v}$ in a matrix Lie algebra $\mathfrak{g}$:
$$\exp(\mathbf{v}) = \sum_{k=0}^{\infty} \frac{\mathbf{v}^k}{k!} \tag{2.1}$$
2. If $\mathbf{u}$ and $\mathbf{v}$ are commutative then $\exp(\mathbf{u}+\mathbf{v}) = \exp(\mathbf{u})\exp(\mathbf{v})$.
3. If $\mathbf{c}$ is an invertible matrix then $\exp(\mathbf{cvc}^{-1}) = \mathbf{c}\exp(\mathbf{v})\mathbf{c}^{-1}$.
4. $\det(\exp(\mathbf{v})) = \exp(\mathrm{trace}(\mathbf{v}))$
5. For any norm, $||\exp(\mathbf{v})|| \leq \exp(||\mathbf{v}||)$.
6. If $\exp(\mathbf{w}) = \exp(\mathbf{u})\exp(\mathbf{v})$ then $\exp(-\mathbf{w}) = \exp(-\mathbf{v})\exp(-\mathbf{u})$.

The idea of defining an inverse of the Lie exponential leads to the idea of the Lie logarithm, defined

$$\begin{aligned} \log : \mathbb{G} &\longrightarrow \mathfrak{g} \\ p &\longmapsto \log(p) = \mathbf{v} \end{aligned}$$

where $\mathbf{v}$ is the tangent vector having p as it exp.

If $\mathbb{G}$ is a matrix Lie group of dimension n, the following properties hold:

1. for all $\mathbf{v}$ in the matrix Lie algebra $\mathfrak{g}$:
$$\log(\mathbf{v}) = \sum_{k=1}^{\infty} (-1)^{k+1} \frac{(\mathbf{v}-I)^k}{k!} \tag{2.2}$$
where I is the identity matrix.

2. For any norm, and for any $n \times n$ matrix $\mathbf{c}$, exists an α such that
$$||\log(I + \mathbf{c}) - \mathbf{c}|| \leq \alpha ||\mathbf{c}||^2$$

3. For any $n \times n$ matrix $\mathbf{c}$ and for any sequence of matrix $\{\mathbf{d}_j\}$ such that $||\mathbf{d}_j|| \leq \alpha/j^2$ it follows:
$$\lim_{k\to\infty} \left(I + \frac{\mathbf{c}}{k} + \mathbf{d}_k\right)^k = \exp(\mathbf{c})$$

Here it may be possible to see the beginning of the problem we have to deal with for the rest of the research, each time passing from the finite dimensional case to the infinite dimensional case.
The domain of the logarithm is the matrix Lie group $\mathbb{G}$ in which only the composition (the matrix product) is defined. Nevertheless it is possible to compute $I + \mathbf{c}$, and this still make sense, and satisfy remarkable properties, when applied to the log. In addition the domain of the exponential is the matrix Lie algebra $\mathfrak{g}$, but the exponential can be nevertheless applied to any matrix, and still be defined.
This can be done thanks to the fact that for matrices, $\mathfrak{g}$ and $\mathbb{G}$ are subset of a bigger algebra, the algebra of invertible matrices $GL(n)$ [KKK08]. In this structure the operation of sum is still defined over the group that in the general case admits only compositions, and the infinite series 2.1, 2.2 are doors to pass from the structure of group to the algebra and vice versa. When presenting the rigid body transformation in chapter 3 we will may open another couple of access doors based on numerical approximations.

When dealing with diffeomorphism in practical application we have to deal also with a theoretical: on one side the the operations in Lie algebra are not compatible with the composition in the Lie group: expression as $e + \mathbf{v}$ contains a sum that is undefined (and therefore meaningless) in the Lie group structure. On the other side the (extremely) continuous nature of diffeomorphisms is not compatible with the (extremely) discrete nature of computers. It is not possible to implement something that maintains any of the property of diffeomorphism in a computer. The only options we have for practical implementations are the vector fields discretized on a d dimensional grid. In the paper of Arsigny [ACPA06a], *scaling and squaring* and *inverse scaling and squaring* are proposed for the computation of exponential and logarithm respectively; these algorithms transform a discretized vector field in another discretized vector field, while theoretical domain and codomain of these transformations are radically different. In addition, as shown in [HOP08] for some parameters of Stationary LDDMM and the diffeomorphic Demons the diffeomorphisms involved do not preserve the signs of the Jacobian determinant, and therefore are not diffeomorphisms anymore.

The passage from finite dimensional case of matrices to infinite dimensional case of diffeomorphisms requires a way to represent diffeomorphisms having only discrete vector fields to deal with, where operations within Lie algebra and Lie group are not anymore compatible. The strategy here proposed is to define an operation in the Lie algebra $\mathfrak{g} = \mathcal{V}(\Omega)$ that reflects the properties of the composition in the corresponding Lie group structure $\mathbb{G} = \mathrm{Diff}(\Omega)$.

The Lie log-composition (because based on the Lie logarithm and Lie exponential maps) is defined here as the inner binary operation on the Lie algebra that reflects the composition on the lie group:

$$\oplus : \mathfrak{g} \times \mathfrak{g} \longrightarrow \mathfrak{g} \tag{2.3}$$
$$(\mathbf{v}_1, \mathbf{v}_2) \longmapsto \mathbf{v}_1 \oplus \mathbf{v}_2 = \log(\exp(\mathbf{v}_1) \circ \exp(\mathbf{v}_2)) \tag{2.4}$$

Following properties holds for the Lie log-composition:

1. $\mathfrak{g}$ with the Lie log-composition $\oplus$ is a local topological non-commutative group (local group for short): if $C_{\mathfrak{g}}$ is the internal cut locus of $\mathfrak{g}$ then:

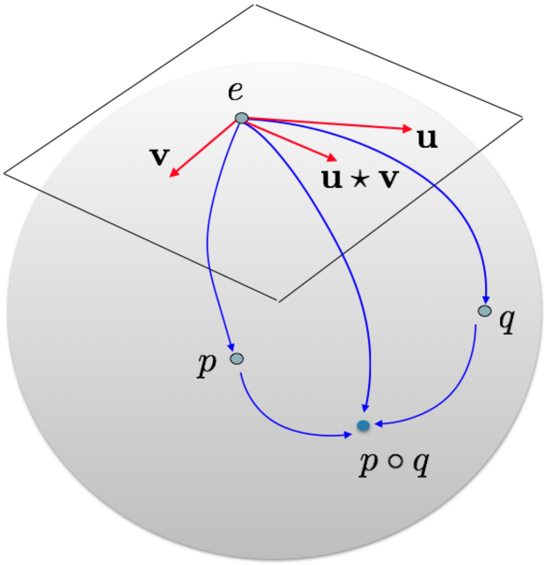

Figure 2.1: graphical visualization of the Lie log-composition.

(a) $(\mathbf{u}_1 \oplus \mathbf{u}_2) \oplus \mathbf{u}_3 = \mathbf{u}_1 \oplus (\mathbf{u}_2 \oplus \mathbf{u}_3)$ for all $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$ in $C_{\mathfrak{g}}$.

(b) $\mathbf{u} \oplus \mathbf{0} = \mathbf{0} \oplus \mathbf{u} = \mathbf{u}$ for all $\mathbf{u}$ in $C_{\mathfrak{g}}$.

(c) $\mathbf{u} \oplus (-\mathbf{u}) = \mathbf{0}$ for all $\mathbf{u}$ in $C_{\mathfrak{g}}$.

2. For all t, s real, such that $(t+s)\mathbf{u}$ is in $C_{\mathfrak{g}}$,

$$(t\mathbf{u}) \oplus (s\mathbf{u}) = (t+s)\mathbf{u}$$

And in particular, if the Lie algebra $\mathfrak{g}$ has dimension 1 the local group structure is compatible with the additive group of the vector space $\mathfrak{g}$.

The algebraic structure $(\mathfrak{g}, \oplus)$ is called Lie log-group. Additional observations on this algebraic structure in the particular case of diffeomorphisms, are proposed in the next chapter. To compute the log-composition there is a formula, the BCH (for matrix Lie group [Hal15], general case [Woj94], applied to medical imaging [VPPA08]), that provides the exact solution to the Log-composition.

$$BCH(\mathbf{u}, \mathbf{v}) = \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u}, \mathbf{v}] + \frac{1}{12}([\mathbf{u}, [\mathbf{u}, \mathbf{v}]] + [\mathbf{v}, [\mathbf{v}, \mathbf{u}]]) - \frac{1}{24}[\mathbf{v}, [\mathbf{u}, [\mathbf{u}, \mathbf{v}]]] + ...$$

This expansion will provide the most immediate way to obtain a numerical computation of $\mathbf{v}_1 \oplus \mathbf{v}_2$.

2.3 Affine Exponential, Affine Logarithm and Parallel Transport: Definition and Properties

Considering a Lie Group $\mathbb{G}$ with a connection ∇ (that provides geodesics and curvature over manifold on which no Riemannian metric has been defined, see [dCV92]), the vector field $\nabla_U(V)$ associates at each point of the manifold the projection on the tangent plane of the covariant derivative of U in the direction of V.
First consequence of the definition of the connection is the possibility of define *geodesics* between points p and q of the manifold. Note that in this case the concept of geodesic does

not involves any metric defined on the surface of the manifold. If also a Riemannian metric is defined on $\mathbb{G}$, then geodesics defined by the metric coincides with the geodesics defined by the connection only for the particular Levi-Civita connection (see [dCV92]).

Another consequence of the definition of connection allows to define a new kind of exponential from the Lie algebra to the Lie group that relies on the concept of geodesics. This time the tangent plane that defines the Lie algebra is considered at the point p of the Lie group and $\mathbf{v} \in T_p\mathbb{G} \simeq \mathfrak{g}$ is a tangent vector at the point p:

$$\begin{aligned} \exp : \mathbb{G} \times \mathfrak{g} &\longrightarrow \mathbb{G} \\ (p, \mathbf{v}) &\longmapsto \exp_p(\mathbf{v}) = \gamma(1; p, \mathbf{v}) \end{aligned}$$

The curve $\gamma(t; p, \mathbf{v}) = \gamma(t)$ on $\mathbb{G}$ is the unique one with the following features:

$$\gamma(0) = p \qquad \dot{\gamma}(0) = \mathbf{v} \qquad \nabla_{\dot{\gamma}}\dot{\gamma} = 0$$

This second kind of exponential is called *affine exponential*.

The inverse of the affine exponential, the affine logarithm is defined as:

$$\begin{aligned} \log : \mathbb{G} \times \mathbb{G} &\longrightarrow T_p\mathbb{G} \simeq \mathfrak{g} \\ (p, q) &\longmapsto \log_p(q) = \mathbf{v} \end{aligned}$$

Where $\mathbf{v}$ is the vector at the tangent plane defined at p such that the curve on $\mathbb{G}$ with the following features

$$\gamma(0) = p \qquad \gamma(1) = q \qquad \nabla_{\dot{\gamma}}\dot{\gamma} = 0$$

has as its tangent in p the vector $\mathbf{v}$.

For further details and properties properties we refer to the literature; here we wish to provide only the intuitive idea that it is possible to move on the fiber bundle of the Lie group transporting in some sense a tangent vector defined at the identity on another tangent space. Certainly the Lie group possess a unique Lie algebra, as the tangent space at some point (the group's identity by convention), but two different tangent space (so two times the same Lie algebra structure) may not be oriented in the same way.

In this section we introduce the concept of parallel transport for the Lie group $\mathbb{G}$. On this definition, again borrowed from differential geometry (for introduction and general definition: [MTW73], [Kne51], [KMN00] and [War13] for a definition of tangent vector field well suited for the parallel transport. For medical imaging applications [LAP11], [PL$^+$11], [LP13], [LP14b]), relies a method for the computation of the log-composition developed in this research for the first time.

Definition 2.3.1. Let $\mathbb{G}$ be a finite dimensional connected Lie group defined with a connection ∇ and V a $\mathcal{C}^\infty$ vector field defined over $\mathbb{G}$. Given $p, q \in \mathbb{G}$ and $\gamma : [0, 1] \to \mathbb{G}$ such that $\gamma(0) = p$ and $\gamma(1) = q$, the vector $V_p \in T_p\mathbb{G}$, is *parallel transported along* γ up to $T_q\mathbb{G}$ if V satisfies

$$\forall t \in [0, 1] \qquad \nabla_{\dot{\gamma}} V_{\gamma(t)} = 0$$

The *parallel transport* is the function that maps V_p from $T_p\mathbb{G}$ to $T_q\mathbb{G}$ along γ:

$$\begin{aligned} \Pi(\gamma)_p^q : T_p\mathbb{G} &\longrightarrow T_q\mathbb{G} \\ V_p &\longmapsto \Pi(\gamma)_p^q(V_p) = V_q \end{aligned}$$

In the next properties we explore how did parallel transport and affine exponential behave when expressed as a composition and when the signs change.

Property 2.3.1 (Inversion)**.** $\mathbb{G}$ Lie group, ∇ connection, $p, q \in \mathbb{G}$. Given γ such that $\gamma(0) = p$, $\gamma(1) = q$ and $\mathbf{u} \in T_p\mathbb{G}$, we have:

1. $\Pi(\gamma)_p^q(-\mathbf{u}) = -\Pi(\gamma)_p^q(\mathbf{u}))$
2. $q = \exp_p(\mathbf{u}) \iff p = \exp_q(-\Pi(\gamma)_p^q(\mathbf{u}))$

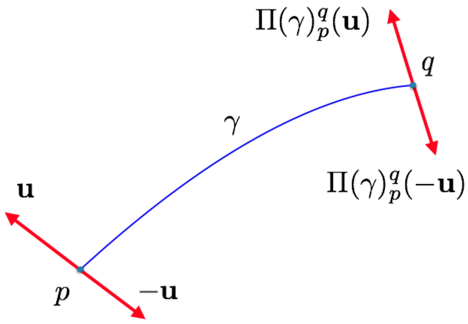

Figure 2.2: First inversion property.

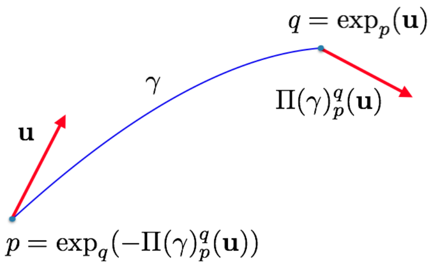

Figure 2.3: Second inversion property.

Property 2.3.2 (change of signs of the composition for affine exponential)**.** $\mathbb{G}$ Lie group, ∇ connection, $p, q \in \mathbb{G}$, $\mathbf{u} \in T_p\mathbb{G}$, $\mathbf{v} \in T_q\mathbb{G}$ and $q = \exp_p(\mathbf{u})$. Let β be the tangent curve to $\mathbf{u}$ at p such that $\beta(1) = q$ and $r = \exp_b(\mathbf{v})$. Given $\mathbf{w} \in T_p\mathbb{G}$ such that

$$\exp_p(\mathbf{w}) = \exp_q(\mathbf{v}) \circ \exp_p(\mathbf{u})$$

Then

$$\exp_p(-\mathbf{w}) = \exp_{\beta(-1)}(-\Pi(\beta)_q^{\beta(-1)}(\mathbf{v})) \circ \exp_p(-\mathbf{u})$$

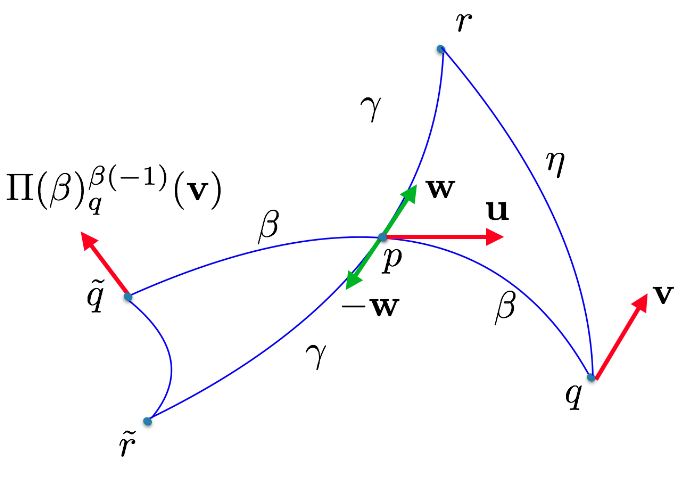

Figure 2.4: Change of sign property.

Lemma 2.3.1. Let $\mathbb{G}$ be a finite dimensional connected Lie group defined with a Cartan connection ∇ and $\mathbf{u}$ tangent vector in $T_e\mathbb{G}$. Let γ be a geodesic defined on $\mathbb{G}$ such that $\gamma(0) = e$, $\dot{\gamma}(0) = \mathbf{u}$ and $p = \gamma(1)$, point in the Lie group. Let β be the curve over $\mathbb{G}$ defined as $\beta(t) = p \circ \gamma(t)$, then the two following conditions hold:

1. If ∇ is a Cartan connection then β is a geodesic.
2. For $\mathbf{u}_p := D(L_p)_e(\mathbf{u}) \in T_p\mathbb{G}$, push forward of the left-translation:
$$\exp_p(t\mathbf{u}_p) = p \circ \exp_e(tD(L_{p^{-1}})_p(\mathbf{u}_p)) = p \circ \exp_e(t\mathbf{u}) \tag{2.5}$$

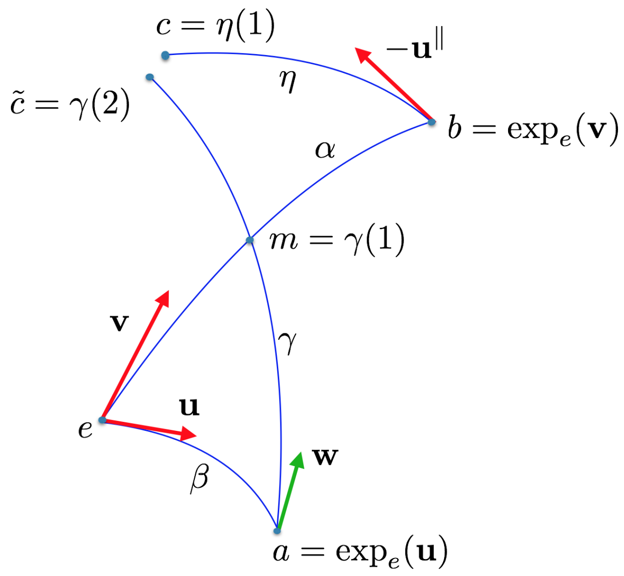

Figure 2.5: Pole ladder applied to parallel transport.

The following theorem is an application of the pole ladder [LAP11] for the computation of the exponential that will underpin one of the numerical methods for the computation of the log-composition.

Theorem 2.3.1. Let $\mathbb{G}$ be a finite dimensional connected Lie group defined with a Cartan connection ∇. If, for each couple of linearly independent vectors $\mathbf{u}, \mathbf{v} \in T_e\mathbb{G}$, the following conditions hold:

$$a = \exp_e(\mathbf{u}) \qquad b = \exp_e(\mathbf{v})$$
$$\mathbf{v}_a^{\|} = \Pi(\gamma)_e^a(\mathbf{v})$$
$$\gamma : [0,1] \to \mathbb{G} \quad \gamma(0) = e \quad \dot{\gamma}(0) = \mathbf{u}$$

$$\mathbf{v}^{\|} := D(L_{a^{-1}})_e(-\Pi(\alpha)_a^e(\mathbf{v}_a^{\|}))$$

Then it follows that if $\mathbf{u}$ is in the open ball $B(\mathbf{0}, \epsilon)$, then exists a $\tilde{\epsilon} > 0$ such that $\mathbf{u}_e^{\|}$ belongs to $B(\mathbf{0}, \tilde{\epsilon})$, and exists a $\delta(\epsilon + \tilde{\epsilon}) > 0$ such that the inequality

$$||\log(\exp_e(\mathbf{v}^{\|})) - \log(\exp_e\big(\frac{\mathbf{u}}{2}\big) \circ \exp_e(\mathbf{v}) \circ \exp_e\big(-\frac{\mathbf{u}}{2}\big))|| < \delta(\epsilon + \tilde{\epsilon})$$

holds.

The last statement can be rewritten as the approximation:

$$\exp_e(\mathbf{v}^{\|}) \simeq \exp_e\big(\frac{\mathbf{u}}{2}\big) \circ \exp_e(\mathbf{v}) \circ \exp_e\big(-\frac{\mathbf{u}}{2}\big) \tag{2.6}$$

that will turn out to be the main tool for the computation of the log-composition using parallel transport.

When considering the equation 2.5, we use implicitly the formula for the change of base for affine exponential and logarithm [APA06]. It is in fact possible, using the derivative of the left-translation L_p, to bring back the exp and the log functions based at the point p of the manifold to the exp and the log evaluated at the identity using the following formulas:

$$\log_p(q) = DL_p(e)\log_e(q) \tag{2.7}$$

$$\exp_p(\mathbf{u}) = p \circ \exp_e(DL_p(e)^{-1}\mathbf{u}) \tag{2.8}$$

Further theoretical developments are beyond the aim of this research, but the reader can refer to the bibliography. In the next section we present the numerical methods for the computation of the log composition.

2.4 Numerical Computations of the Log-composition

In this section we provide explicit formulas for the computation of the log composition:

$$\mathbf{v}_1 \oplus \mathbf{v}_2 = \log(\exp(\mathbf{v}_1) \circ \exp(\mathbf{v}_2)) \tag{2.9}$$

using the tools introduced in the previous sections.

2.4.1 Truncated BCH formula for the Log-composition

To compute the Lie log-composition, literature provides the BCH formula (for matrices [Hal15], general case: [KO89] , [Ser09]), defined as the solution of the equation $\exp(\mathbf{w}) = \exp(\mathbf{u}) \circ \exp(\mathbf{v})$, for $\mathbf{u}$ and $\mathbf{v}$ in the Lie algebra $\mathfrak{g}$:

$$BCH(\mathbf{u},\mathbf{v}) = \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}([\mathbf{u},[\mathbf{u},\mathbf{v}]] + [\mathbf{v},[\mathbf{v},\mathbf{u}]]) - \frac{1}{24}[\mathbf{v},[\mathbf{u},[\mathbf{u},\mathbf{v}]]] + ... \tag{2.10}$$

It consists of an infinite series of Lie bracket whose asymptotic behaviour cannot be predicted only from the coefficient of each nested Lie bracket term. In practical applications it can be computed using its *approximation of degree k*, defined as the sum of the BCH terms having no more than k nested Lie bracket. For example:

$$\begin{aligned}
BCH^0(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} \\
BCH^1(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] \\
BCH^2(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}([\mathbf{u},[\mathbf{u},\mathbf{v}]] + [\mathbf{v},[\mathbf{v},\mathbf{u}]]) \\
BCH^3(\mathbf{u},\mathbf{v}) &= \mathbf{u} + \mathbf{v} + \frac{1}{2}[\mathbf{u},\mathbf{v}] + \frac{1}{12}([\mathbf{u},[\mathbf{u},\mathbf{v}]] + [\mathbf{v},[\mathbf{v},\mathbf{u}]]) - \frac{1}{24}[\mathbf{v},[\mathbf{u},[\mathbf{u},\mathbf{v}]]]
\end{aligned}$$

These numerical approximations of the log-composition $\mathbf{u} \oplus \mathbf{v}$ can be considered as a first step but are not satisfactory since they do not provide any condition to consider the error carried by each term.

2.4.2 Taylor Expansion Method for the Log-composition

A more sophisticated numerical method to manage the nested Lie brackets for the computation of the log-composition is based on the Taylor expansion.
As shown in the appendix of [KO89] the terms of the BCH can be recollected using the Hausdorff method: each of the therm containing the n-th power of the vector $\mathbf{v}$ are collected together in the formal series V^n. Therefore

$$BCH(\mathbf{u},\mathbf{v}) = \mathbf{u} + V^1\mathbf{v} + V^2\mathbf{v} + V^3\mathbf{v} + \cdots$$

Given the adjoint map:

$$\begin{aligned} ad_{\mathbf{u}} : \mathfrak{g} &\longrightarrow \mathfrak{g} \\ \mathbf{v} &\longmapsto ad_{\mathbf{u}}\mathbf{v} := [\mathbf{u}, \mathbf{v}] \end{aligned}$$

and the multiple adjoint maps, defined as:

$$\mathrm{ad}^n_{\mathbf{u}}\mathbf{v} := [\underbrace{\mathbf{u}, [\mathbf{u}, ...[\mathbf{u}}_{\text{n-times}}, \mathbf{v}]...]]$$

$$\mathrm{ad}^{-n}_{\mathbf{u}}\mathbf{v} := [[...[\mathbf{v}, \underbrace{\mathbf{u}]...], \mathbf{u}}_{\text{n-times}}] = (-1)^n \mathrm{ad}^n_{\mathbf{u}}\mathbf{v}$$

it can be proved that hen the operator V^1 is applied to $\mathbf{v}$, it provides the linear part of $\mathbf{v}$ in the BCH formula. It results

$$V_1 = \frac{\mathrm{ad}^{-1}_{\mathbf{u}}}{\exp(\mathrm{ad}_{\mathbf{u}}) - 1} = \sum_{n=0}^{\infty} \frac{(-1)^n B_n}{n!} \mathrm{ad}^{-n}_{\mathbf{u}} = \sum_{n=0}^{\infty} \frac{B_n}{n!} \mathrm{ad}^n_{\mathbf{u}}$$

where $\{B_n\}_{n=0}^{\infty}$ is the sequence of the second-kind Bernoulli number. If first-kind Bernoulli number are used, then each term of the summation must be multiplied for $(-1)^n$, as did for example in [KO89]. The denominator is defined within the structure of the formal power series ring [MT13].
In conclusion, the log-composition can expressed as:

$$\mathbf{u} \oplus \mathbf{v} = \mathbf{u} + \frac{\mathrm{ad}^{-1}_{\mathbf{u}}}{\exp(\mathrm{ad}_{\mathbf{u}}) - 1}\mathbf{v} + \mathcal{O}(\mathbf{v}^2)$$

$$\mathbf{u} \oplus \mathbf{v} = \mathbf{u} + \sum_{n=0}^{\infty} \frac{B_n}{n!} \mathrm{ad}^n_{\mathbf{u}}\mathbf{v} + \mathcal{O}(\mathbf{v}^2) \tag{2.11}$$

that will turn out to be an important tool for the computation of the log-composition in the finite dimensional case.

2.4.3 Parallel Transport Method for the Log-composition

To obtain a numerical computation for the log-composition using parallel transport, we have to consider two assumptions:

1. If $\mathbf{v}^{\parallel}$ is defined as in theorem 2.3.1, then exists an $\hat{\epsilon} > 0$ such that

$$||\mathbf{u} \oplus \mathbf{v} - (\mathbf{u} + \mathbf{v}^{\parallel})|| < \hat{\epsilon}$$

2. If the vector $\mathbf{w} \in \mathfrak{g}$ is small enough, then:

$$\exp(\mathbf{u}) \simeq e + \mathbf{u}$$

The first assumption is a consequence of geometrical intuition, while the second one is valid if the hypothesis of proposition 8.6 pag. 163 [You10] holds. A deepening in this direction is not in the scope of this research. For our purposes we will consider the vector $\mathbf{w}$ small enough to make the approximation questioned here reasonable.

From these assumptions and from equation 2.6 it follows that

$$\mathbf{u} \oplus \mathbf{v} \simeq \mathbf{u} + \mathbf{v}^{\|}$$
$$e + \mathbf{v}^{\|} \simeq \exp_e\big(\frac{\mathbf{u}}{2}\big) \circ \exp_e(\mathbf{v}) \circ \exp_e\big(-\frac{\mathbf{u}}{2}\big)$$

Therefore

$$\mathbf{u} \oplus \mathbf{v} \simeq \mathbf{u} + \exp_e\big(\frac{\mathbf{u}}{2}\big) \circ \exp_e(\mathbf{v}) \circ \exp_e\big(-\frac{\mathbf{u}}{2}\big) - e \tag{2.12}$$

With the truncated BCH and the Taylor expansion, this is the third numerical method for the computation of the log-composition explored in this thesis.
We have to notice that when we apply it on the infinite dimensional case, the approximation 2.12 holds under the following additional assumption:

3. Theorem 2.3.1 holds when the Lie group is infinite dimensional.

An eventual confirmation is at the moment not known to the author. We assume it is true in coherence with what has been said in the introduction, section 1.1.3.

Chapter 3

Spatial Transformations for the Computations of the Log-composition: SE(2) and SVF

Every working mathematician knows that if one does not control oneself (best of all by examples), then after some ten pages half of all the signs in formulae will be wrong and twos will find their way from denominators into numerators.
-V.I. Arnold

In the previous chapter we have introduced some essential mathematical tools for the numerical computation of the log-composition. Each of the theoretical elements depends strongly on the transformations considered, in this chapter we will see how they can be applied for the cases of SE(2) and the Lie group of diffeomorphisms parametrized with stationary velocity fields:

$SE(2)$ - The group of rigid body transformation of the plane (any combination of bi-dimensional rotations and translations) is a good playground to test the numerical methods introduced so far, since results can be compared with a closed form. A representation of this Lie group as a subgroup of the general linear group $GL(2)$, with corresponding Lie algebra will be provided, with closed form for the log-computation.

SVF - The subgroup of the set of diffeomorphisms parametrized by SVF, is the second group utilized to test the numerical methods here presented for the computation of the log-composition. In this case we do not know any closed form, but if we consider an "improper norm" in the space of transformation we still have a method to compare SVF and assess the quality of the results.

3.1 The Lie Group of Rigid Body Transformations

Each element of the group of rigid body transformation (or euclidean group) $SE(2)$ can be computed as the consecutive application of a rotation and a translation applied to any point $(x, y)^T$ of the plane:

$$\begin{pmatrix} X \\ Y \end{pmatrix} = R(\theta) \begin{pmatrix} x \\ y \end{pmatrix} + t = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t^x \\ t^y \end{pmatrix}$$

where the rotation matrix defined by θ belongs to the special orthogonal group $SO(2)$.
We can represent the elements of $SE(2)$ in two different form: as ternary vector (restricted form)

$$SE(2)^v := \{(\theta, t^x, t^y) \mid \theta \in [0, 2\pi), t^x, t^y \in \mathbf{R}^2\}$$

or with matrices (matrix form)

$$SE(2) := \left\{ \begin{pmatrix} R(\theta) & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & t^x \\ \sin(\theta) & \cos(\theta) & t^y \\ 0 & 0 & 1 \end{pmatrix} \mid \theta \in [0, 2\pi), (t^x, t^y) \in \mathbf{R}^2 \right\}$$

The group $SE(2)$ it is a manifold with a differentiable structure compatible with the operation of composition, whose Lie algebra is given in matrix form by (see [Hal15, Gal11] for an introduction).

$$\mathfrak{se}(2) := \left\{ \begin{pmatrix} dR(\theta) & dt \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\theta & dt^x \\ \theta & 0 & dt^y \\ 0 & 0 & 0 \end{pmatrix} \mid \theta \in [0, 2\pi), (tx, ty) \in \mathbf{R}^2 \right\}$$

and it is indicated with $\mathfrak{se}(2)^v$ in its restricted form.

Given r, element of $SE(2)$ with $\theta \neq 0$. Its image with the Lie group logarithm is

$$\begin{aligned} \log(r) &= \sum_{k=1}^{\infty} (-1)^{k+1} \frac{(r-I)^k}{k} = \begin{pmatrix} dR(\theta) & L(\theta)t \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & -\theta & \frac{\theta}{2}(\frac{\sin(\theta)}{1-\cos(\theta)} t^x + t^y) \\ \theta & 0 & \frac{\theta}{2}(-t^x + \frac{\sin(\theta)}{1-\cos(\theta)} t^y) \\ 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

where therefore

$$dR(\theta) = \begin{pmatrix} 0 & -\theta \\ \theta & 0 \end{pmatrix} \qquad L(\theta) = \frac{\theta}{2} \begin{pmatrix} \frac{\sin(\theta)}{1-\cos(\theta)} & 1 \\ -1 & \frac{\sin(\theta)}{1-\cos(\theta)} \end{pmatrix}$$

On the way back, the exponential of $dr \in \mathfrak{se}(2)$ is given by:

$$\begin{aligned} \exp(dr) &= \sum_{k=1}^{\infty} \frac{dr^k}{k!} = \begin{pmatrix} R(\theta) & L(\theta)^{-1} t \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos(\theta) & -\sin(\theta) & \frac{1}{\theta}(\sin(\theta) dt^x - (1-\cos(\theta)) dt^y) \\ \sin(\theta) & \cos(\theta) & \frac{1}{\theta}(-(1-\cos(\theta)) dt^x + \sin(\theta) dt^y) \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

where

$$L(\theta)^{-1} = \frac{1}{\theta} \begin{pmatrix} \sin(\theta) & -(1-\cos(\theta)) \\ (1-\cos(\theta)) & \sin(\theta) \end{pmatrix}$$

When θ is zero, $R(\theta)$ and $dR(\theta)$ coincide with the identity, and the transformation results in a translation. For proof and further details see for example [Gal11] [Hal15].

At this point it is important to notice that:

1. The infinite series of matrices do not raises any theoretical issues, since the sum is defined in the group as subset of a bigger algebra that contains both the Lie group and

the Lie algebra. It appears to be the natural way to move back and forth from the group to the algebra. A second door to passing from one structure to the other, when the rotation θ is little is provided by the following approximations:

$$\exp(r) \simeq I + r \qquad \log(dr) \simeq dr - I$$

In fact for little θ, $\sin(\theta) \simeq \theta$, $\cos(\theta) \simeq 0$ and $L(\theta) \simeq I$.

2. The map exp is not well defined over its whole domain $\mathfrak{se}(2)$. Given two elements $(\theta_0, dt_0^x, dt_0^y)$ and $(\theta_1, dt_1^x, dt_1^y)$, they have the same image with exp function if the two following conditions are both satisfied:

   i) Exists an integer k such that $\theta_0 = \theta_1 + 2k\pi$.
   ii) the translation (dt_0^x, dt_0^y) coincides with (dt_1^x, dt_1^y) up to a factor $\frac{\theta_0 \mod 2\pi}{\theta_1}$.

   To have a bijective correspondence we have to restrict the domain of exp to a space where if $\exp(\theta_0, dt_0^x, dt_0 y) = \exp(\theta_1, dt_1^x, dt_1^y)$ implies $(\theta_0, dt_0^x, dt_0^y) = (\theta_1, dt_1^x, dt_1^y)$. It can be easy to prove that the sought space is the quotient of $\mathfrak{se}(2)$ over the equivalence relation $\sim$, defined as

$$(\theta_0, dt_0^x, dt_0^y) \sim (\theta_1, dt_1^x, dt_1^y)$$
$$\overset{\text{def}}{\iff}$$
$$\exists k \in \mathbb{Z} \mid \theta_0 = \theta_1 + 2k\pi \;\text{ and }\; (dt_0^x, dt_0^y) = \frac{\theta_0 \mod 2\pi}{\theta_1}(dt_1^x, dt_1^y)$$

   The new algebra defined by the set of equivalence classes of this relation is indicated - with the standard convention, see [Art11] - with $\mathfrak{se}(2)/\sim$. With this restriction of the domainexp is a bijection having log as its inverse. What said so far can be summarize in the following commutative diagram:

$$\begin{tikzcd}
\mathfrak{se}(2) \arrow[dr, two heads, "\pi"] & \\
 & \mathfrak{se}(2)/\sim \arrow[dl, "\exp"] \\
SE(2) \arrow[uu, "\log"] &
\end{tikzcd}$$

   and with the schematic figure 3.1.

3.1.1 Computations of Log-composition in $\mathfrak{se}(2)$

The log-composition of two elements $dr_0 = (\theta_0, dt_0^x, dt_0^y)$ ans $dr_1 = (\theta_1, dt_1^x, dt_1^y)$ of $\mathfrak{se}(2)/\sim$ results, in consequence of [**?**] as

$$dr_0 \oplus dr_1 = \log(\exp(dr_0) \circ \exp(dr_1)) \tag{3.1}$$

The approximations of the log-composition using truncated BCH formulas are straightforward:

$$dr_0 \oplus dr_1 \simeq BCH^0(dr_0, dr_1) = dr_0 + dr_1$$
$$dr_0 \oplus dr_1 \simeq BCH^1(dr_0, dr_1) = dr_0 + dr_1 + \frac{1}{2}[dr_0, dr_1]$$
$$dr_0 \oplus dr_1 \simeq BCH^2(dr_0, dr_1) = dr_0 + dr_1 + \frac{1}{2}[dr_0, dr_1] + \frac{1}{12}([dr_0, [dr_0, dr_1]] + [dr_1, [dr_1, dr_0]])$$

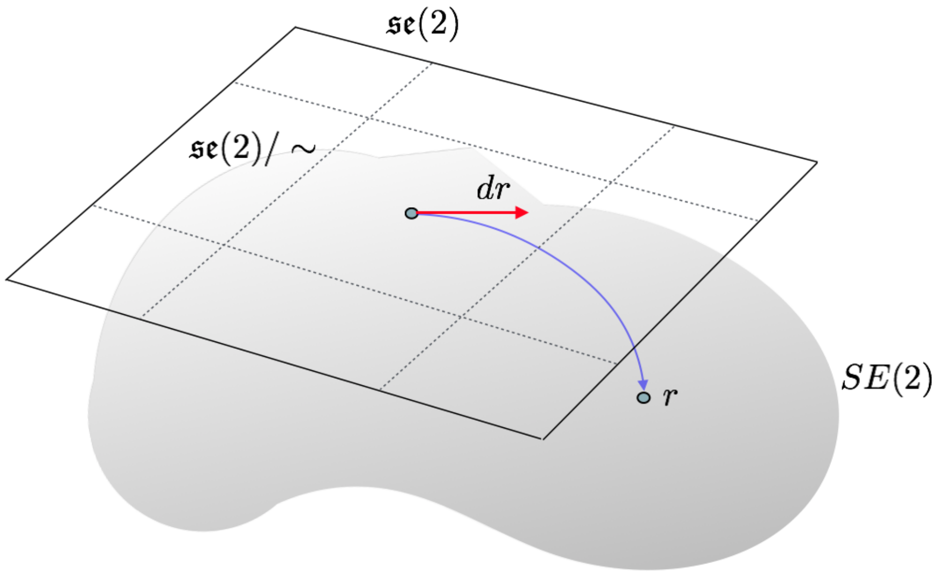

Figure 3.1: The Lie algebra $\mathfrak{se}(2)/\sim$ defined as the quotient of the Lie algebra $\mathfrak{se}(2)$ over the equivalence relation $\sim$ is in bijective correspondence with $SE(2)$.

To compute the approximation with the Taylor method, and so to compute the equation 2.11, we observe that the restricted form of the Lie bracket is given by

$$\begin{aligned}[dr_0, dr_1] &= (0, dR(\theta_0)dt_1 - dR(\theta_1)dt_0)^T \\ &= (0, -\theta_0 dt_1^y + \theta_1 dt_0^y, \theta_0 dt_1^x - \theta_1 dt_0^x)^T\end{aligned}$$

Therefore, the adjoint operator can be written in matrix form as a dual matrix of dr:

$$\mathrm{ad}_{dr} = \begin{pmatrix} 0 & 0 & 0 \\ dt^y & 0 & -\theta \\ -dt^x & \theta & 0 \end{pmatrix}$$

In fact, when applied to dr_1 it results in the Lie bracket:

$$\mathrm{ad}_{dr_0} dr_1 = \begin{pmatrix} 0 & 0 & 0 \\ dt_0^y & 0 & -\theta_0 \\ -dt_0^x & \theta_0 & 0 \end{pmatrix} \begin{pmatrix} \theta_1 \\ dt_1^x \\ dt_1^y \end{pmatrix} = \begin{pmatrix} 0 \\ -\theta_0 dt_1^y + \theta_1 dt_0^y \\ \theta_0 dt_1^x - \theta_1 dt_0^x \end{pmatrix}$$

To compute the Taylor approximation proposed in equation 2.11 of the log composition, indicating $dt^\star = (dt^y, -dt^x)$ it can be proved easily by induction that

$$\mathrm{ad}_{dr}^n = \begin{pmatrix} 0 & 0 \\ dt^\star & dR(\theta) \end{pmatrix}^n = \begin{pmatrix} 0 & 0 \\ dR(\theta)^{n-1} dt^\star & dR(\theta)^n \end{pmatrix}$$

And so the series involved in the equation 2.11 become

$$\sum_{n=0}^{\infty} \frac{B_n}{n!} \mathrm{ad}_{dr}^n = \sum_{n=0}^{\infty} \frac{B_n}{n!} \begin{pmatrix} 0 & 0 \\ dR(\theta)^{n-1} dt^\star & dR(\theta)^n \end{pmatrix}$$

We can split it in two part, the rotational part $dR(\theta)^n$ and the translational part $dR(\theta)^{n-1}dt^\star$. The rotational part, using the nature of Bernoulli numbers and its generative equation, when

$\theta \neq 0$ become

$$\begin{aligned}
\sum_{n=0}^{\infty} \frac{B_n}{n!} dR(\theta)^n &= I + \frac{1}{2} dR(\theta) + \sum_{n=1}^{\infty} \frac{B_{2n}}{2n!} dR(\theta)^{2n} \\
&= I + \frac{1}{2} dR(\theta) + (\sum_{n=1}^{\infty} \frac{B_{2n}}{2n!} (i\theta)^{2n}) I \\
&= \frac{1}{2} dR(\theta) + (\sum_{n=0}^{\infty} \frac{B_n}{n!} (i\theta)^n - \frac{1}{2} i\theta) I \\
&= \frac{1}{2} dR(\theta) + (\frac{i\theta e^{i\theta}}{e^{i\theta} - 1} - \frac{1}{2} i\theta) I \\
&= \frac{1}{2} dR(\theta) + \frac{\theta/2}{\tan(\theta/2)} I
\end{aligned}$$

where the equation $dR(\theta)^{2n} = (i\theta)^{2n} I$. For the translational part we have

$$\begin{aligned}
\sum_{n=1}^{\infty} \frac{B_n}{n!} dR(\theta)^{n-1} dt^\star &= dR(\theta)^{-1} \Big(\sum_{n=1}^{\infty} \frac{B_n}{n!} dR(\theta)^n \Big) dt^\star \\
&= dR(\theta)^{-1} \Big(\sum_{n=0}^{\infty} \frac{B_n}{n!} dR(\theta)^n - I \Big) dt^\star \\
&= dR(\theta)^{-1} \Big(\sum_{n=0}^{\infty} \frac{1}{2} dR(\theta) + \frac{\theta/2}{\tan(\theta/2)} I - I \Big) dt^\star \\
&= dR(\theta)^{-1} \Big(\sum_{n=0}^{\infty} \frac{1}{2} dR(\theta) + \frac{\theta/2}{\tan(\theta/2)} I - I \Big) dt^\star \\
&= \Big(\frac{1}{2} I + (\frac{\theta/2}{\tan(\theta/2)} - 1) dR(\theta)^{-1} \Big) dt^\star
\end{aligned}$$

Finally the closed form for the Taylor approximation of the log-composition is [Ver14]:

$$dr_0 \oplus dr_1 = dr_0 + \sum_{n=0}^{\infty} \frac{B_n}{n!} \mathrm{ad}_{dr_0}^n dr_1 + \mathcal{O}(dr_1^2) = dr_0 + \mathbf{J}(dr_0, dr_1) dr_1 + \mathcal{O}(dr_1^2) \tag{3.2}$$

where

$$\mathbf{J}(dr_0, dr_1) = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{\theta_0/2 - \tan(\theta_0/2)}{\theta_0 \tan(\theta_0/2)} dt_0^x + \frac{1}{2} dt_0^y & \frac{\theta_0/2}{\tan(\theta_0/2)} & -\theta_0/2 \\ -\frac{1}{2} dt_0^x - \frac{\theta_0/2 - \tan(\theta_0/2)}{\theta_0 \tan(\theta_0/2)} dt_0^y & \theta_0/2 & \frac{\theta_0/2}{\tan(\theta_0/2)} \end{pmatrix}$$

The approximation of the log-composition using parallel transport is a straightforward application of the equation 2.12:

$$dr_0 \oplus dr_1 \simeq dr_0 + \exp\big(\frac{dr_0}{2}\big) \exp(dr_1) \exp\big(-\frac{dr_0}{2}\big) - I \tag{3.3}$$

where the composition in the Lie group coincides with the product of matrix in the bigger algebra $GL(3)$ that contains both the Lie group $SE(2)$ and the Lie algebra $\mathfrak{se}(2)$.

3.2 One-parameter Subgroup Generated by Stationary Velocity Fields

As previously said in section 1.1.3, the passage from the finite to the infinite dimensional case is not free of deceptions. For the particular case of diffeomorphisms over the compact

subset $\Omega \subset \mathbb{R}^d$, indicated with $\mathit{Diff}(\Omega)$, the exponential map is not a local bijection (see the counterexample in [LP13], pag. 6 or the definition of Koppel-diffeomorphisms [Gra88] pag. 115). We will consider only the subset $\mathit{Diff}^1(\Omega)$ of $\mathit{Diff}(\Omega)$ containing only the diffeomorphisms that are in the image of the exponential function exp. In other world, $\mathit{Diff}^1(\Omega)$ is the set the diffeomorphisms that belongs to the one-parameter subgroup.
Each of its element are generated by a tangent vector field in $\mathcal{C}^\infty(\Omega)$ (see definitions in subsection 1.2.2) through an ordinary differential equation (see for example [Arn06]). These generating tangent vector field can be divided into two classes, as well as the related ODE:

1. Stationary - or homogeneous
$$\frac{d\varphi(t)}{dt} = V_{\varphi(t)}$$
2. Non-stationary - or non-homogeneous
$$\frac{d\varphi(t)}{dt} = V_{(t,\varphi(t))}$$

For a fixed t both the stationary vector field (or SVF) $V_{\varphi(t)}$ and the time varying vector field (or TVVF) $V_{(t,\varphi(t))}$ have solutions that belong to $\mathit{Diff}^1(\Omega)$. When varying t, the TVVFs behave as continuous paths over the tangent vector field, corresponding to the one-parameter subgroup $\mathit{Diff}^1(\Omega)$ through the map exp.
Thanks to the Cauchy theorem that ensures the uniqueness of the solution of the ODE under the initial condition $\varphi(0) = e$, the local bijection between $\mathit{Diff}^1(\Omega)$ and the Lie algebra of the tangent vector fields over Ω is ensured (see [Mil82], [KW08]):
$$\mathcal{V}(\Omega) \simeq \mathit{Diff}^1(\Omega) \subset \mathrm{Diff}(\Omega)$$
and, as consequence of the definition of exp and log it follows that
$$\mathit{Diff}^1(\Omega) = \exp(\mathcal{V}(\Omega))$$
$$\mathcal{V}(\Omega) = \log(G)$$
In the LDDMM framework (see section 1.2.2) only TVVFs are considered, while after subsequent paper of [ACPA06a] the attention has been restricted to SVF for practical applications. For our purposes when talking about diffeomorphisms, we will take into account only the one embedded in the one-parameter subgroup $\mathit{Diff}^1(\Omega)$. In other world, the image of a SVF through the exp map.

3.2.1 Computations of Log-composition for SVF

A closed-form for the Taylor Expansion method 2.4.2 to compute the log-composition with element in $\mathit{Diff}^1(\Omega)$ is not known. We will therefore compare the truncated BCH formula with the parallel transport method 2.3. We will exploit the parametrization of discretized SVF using matrices to have a ground truth to compare results. Norm will be computed in the group as the L_2 norm of matrices that represents the SVF.
Given $\mathbf{u}$ and $\mathbf{v}$ in $\mathit{Diff}^1(\Omega)$, $\mathbf{w}_{\text{ground}} = \mathbf{u} \oplus \mathbf{v}$ solution of the log-composition and $\mathbf{w}_{\text{app}}$ its approximation using a numerical method, then their difference is computed in the group as:
$$\text{error} = ||\exp(\mathbf{w}_{\text{ground}}) - \exp(\mathbf{w}_{\text{app}})||_{L^2}$$
where $\exp(\mathbf{w}_{\text{ground}})$ is computed as the composition of the exponentials of $\mathbf{u}$ and $\mathbf{v}$:
$$\exp(\mathbf{w}_{\text{ground}}) = \exp(\mathbf{u}) \circ \exp(\mathbf{v})$$
As previously said, the norm L^2 is considered improperly in a group structure. It can be done only thanks to the fact that the discrete SVF and the corresponding diffeomorphisms $\mathit{Diff}^1(\Omega)$ are implemented with 5-dimensional matrices (see equation 1.3).

Chapter 4

Log-Algorithm using Log-composition

I think you might do something better with the time
than wasting it in asking riddles that have no answers.
-*Alice in Wonderland.*

The *logarithm computation problem* can be stated as follows:

Given p *in a Lie group* $\mathbb{G}$*,*
what is the element $\mathbf{u}$ *in its Lie algebra* $\mathfrak{g}$
such that $\exp(\mathbf{u}) = p$ *?*

There are several numerical methods to compute the approximation of the problem's solution. Arsigny, who first pointed the applications of the Lie logarithm in medical image registration in[AFPA06] and [APA06], proposed the Inverse scaling and squaring. Here we are interested in the numerical iterative algorithm for the computation of the Lie logarithm, called here *log-algorithm*, presented for the first time in [BO08]. In this chapter we present a strong relation between the log-algorithm and the log-composition: in consequence of this, each numerical methods presented in these pages can be applied to find a numerical method to solve the logarithm computation problem.
The first step toward this direction is to introduce the space of the approximations of a Lie algebra and a the Lie group.

4.1 Spaces of Approximations

As seen in section 3.1 of the previous chapter for the particular case of $SE(2)$, if the matrix dr is small enough we can approximate $\exp(dr)$ with $1 + dr$. Aim of this section is to generalize the same approximation for the SVF.
We define two approximating functions:

$$\begin{aligned} \mathrm{app} : \mathfrak{g} &\longrightarrow \mathfrak{g}^{\sim} \\ \mathbf{u} &\longmapsto \exp(\mathbf{u}) - 1 \end{aligned}$$

$$\begin{aligned} \mathrm{App} : \mathbb{G} &\longrightarrow \mathbb{G}^{\sim} \\ \exp(\mathbf{u}) &\longmapsto 1 + \mathbf{u} \end{aligned}$$

Where $\mathfrak{g}^{\sim}$ is the space of approximations of elements of $\mathfrak{g}$, and $\mathbb{G}^{\sim}$ is the space of approximations of elements in $\mathbb{G}$, defined as

$$\begin{aligned}\mathfrak{g}^{\sim} &:= \{\exp(\mathbf{u}) - 1 \mid \mathbf{u} \in \mathfrak{g}\} \cup \mathfrak{g} \\ \mathbb{G}^{\sim} &:= \{1 + \mathbf{u} \mid \mathbf{u} \in \mathbb{G}\} \cup \mathbb{G}\end{aligned}$$

In general $\mathfrak{g}^{\sim} \neq \mathfrak{g}$ and $\mathbb{G}^{\sim} \neq \mathbb{G}$, but in the considered cases of $\mathfrak{se}(2)$ and SVF, when $\mathbf{u}$ is *small enough* it follows that $\exp(\mathbf{u}) - 1 \in \mathfrak{g}$ and $1 + \mathbf{u} \in \mathbb{G}$. Therefore the elements of $\mathfrak{g}^{\sim}$ are compatible with all of the operations of Lie algebra $\mathfrak{g}$ and the elements of $\mathbb{G}^{\sim}$ are compatible with all of the operations of Lie group $\mathbb{G}$.
Lets examine what does *small enough* means in these two cases:

$\mathfrak{se}(2)$ - Since $\mathfrak{se}(2)$ and $SE(2)$ are subset of the bigger algebra $SE(2)$ then exp and log can be defined as infinite series. From

$$\exp(\mathbf{u}) = I + \mathbf{u} + O(\mathbf{u}^2)$$

It follows that $\text{app}(\mathbf{u}) - \mathbf{u} = O(\mathbf{u}^2)$. Thus for all $\mathbf{u}$ smaller than δ for any norm, exists $M(\delta)$ such that

$$||\text{app}(\mathbf{u}) - \mathbf{u}|| < M(\delta)||\mathbf{u}^2||$$

SVF - In case of SVF we do not have any Taylor series and big-O notation available but, according to the proposition 8.6 at page 163 of [You10], if $\mathbf{u}$ is, for any norm, smaller than $\epsilon < 1/C$, where C is the Lipschitz constant of the same norm, then $e + \mathbf{u}$ is a diffeomorphism. With this condition holds that $\text{SVF}^{\sim} = \text{SVF}$.

Therefore, for each small enough $\mathbf{u}$ in $\mathfrak{se}(2)$ or SVF, and considering the definition of the log-composition (equation 2.3) the following properties holds:

1. The approximations $\mathbf{u} \simeq \text{app}(\mathbf{u})$, $\exp(\mathbf{u}) \simeq \text{App}(\exp(\mathbf{u}))$ are meaningful.
2. $\mathbf{u} = \mathbf{v} \oplus (-\mathbf{v} \oplus \mathbf{u})$
3. $\text{app}(\mathbf{v} \oplus \mathbf{u}) = \exp(\mathbf{v})\exp(\mathbf{u}) - 1 \in \mathfrak{g}^{\sim}$

With this machinery, we can finally reformulate the algorithm presented in [BO08] for the numerical computation of the Lie logarithm map using the log-composition.

4.2 The Log-computation Algorithm using Log-composition

If the goal is to find $\mathbf{u}$ when its exponential is known, we can consider the sequence transformations $\{\mathbf{u}_j\}_{j=0}^{\infty}$ that approximate $\mathbf{u}$ as consequence of

$$\mathbf{u} = \mathbf{u}_j \oplus (-\mathbf{u}_j \oplus \mathbf{u}) \Longrightarrow \mathbf{u} \simeq \mathbf{u}_j \oplus \text{app}(-\mathbf{u}_j \oplus \mathbf{u})$$

This suggest that a reasonable approximation for the $(j+1)$-th element of the series can be defined by

$$\mathbf{u}_{j+1} := \mathbf{u}_j \oplus \text{app}(-\mathbf{u}_j \oplus \mathbf{u})$$

If we chose the initial value $\mathbf{u}_0$ to be zero, then the algorithm presented in [BO08] become:

$$\begin{cases}\mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j \oplus \text{app}(-\mathbf{u}_j \oplus \mathbf{u})\end{cases} \tag{4.1}$$

Each strategy that we have examined to compute the Lie composition, become a numerical method for the computation of the logarithm.

4.2.1 Truncated BCH Strategy

At each step, we compute the approximation $\mathbf{v}_{j+1}$ with the k-th truncation of the BCH formula:

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathrm{BCH}^k(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) \end{cases} \tag{4.2}$$

For $k = 0$, the approximation $\mathbf{u}_{j+1}$ results simply the sum $\mathbf{u}_j + \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})$. When $k = 1$, it results

$$\begin{aligned} \mathrm{BCH}^1(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) &= \mathbf{u}_j + \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) \\ &= \mathbf{u}_j + \exp(-\mathbf{u}_j)\exp(\mathbf{u}) - 1 \end{aligned}$$

And $k = 2$ it follows

$$\begin{aligned} \mathrm{BCH}^2(\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})) &= \mathbf{u}_j + \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) + \frac{1}{2}[\mathbf{u}_j, \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u})] \\ &= \mathbf{u}_j + \exp(-\mathbf{u}_j)\exp(\mathbf{u}) - 1+ \\ &+ \frac{1}{2}(\mathbf{u}_j(\exp(-\mathbf{u}_j)\exp(\mathbf{u}) - 1) - (\exp(-\mathbf{u}_j)\exp(\mathbf{u}) - 1)\mathbf{u}_j) \end{aligned}$$

The following theorem presented in [BO08], provides an error bound when $k = \infty$ so when the BCH formula is used, instead one of its truncation.

Theorem 4.2.1 (Bossa)**.** The iterative algorithm

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j \oplus \mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) \end{cases} \tag{4.3}$$

converges to $\mathbf{v}$ with error $\delta_n \in \mathbb{G}$, where

$$\delta_n := \log(\exp(\mathbf{v}) \circ \exp(-\mathbf{v}_n)) \in O(||p - e||^{2^n})$$

We observe that this upper limit can be computed only when a closed-form for the log-composition is available, as for example $\mathfrak{se}(2)$.

4.2.2 Parallel Transport Strategy

If we apply the parallel transport method for the computation of the log-composition, we obtain another version of the log-algorithm:

$$\begin{cases} \mathbf{u}_0 = \mathbf{0} \\ \mathbf{u}_t = \mathbf{u}_{t-1} + \exp(\frac{\mathbf{u}_{t-1}}{2}) \circ \exp(\delta\mathbf{u}_{t-1}) \circ \exp(-\frac{\mathbf{u}_{t-1}}{2}) - e \end{cases} \tag{4.4}$$

We notice that mixing the operation of composition, sum and scalar product makes sense when the involved vectors are *small enough*, as stated in 4.1. Analytical computation of an upper bound error is not straightforward in this case. See section 5.4 for further details and other possible researches.

4.2.3 Symmetrization Strategy

The algorithm 4.1 could have been reformulated alternatively as $\mathbf{u}_{j+1} = \mathrm{app}(\mathbf{u} \oplus -\mathbf{u}_j) \oplus \mathbf{u}_j$. The log-composition is not symmetric therefore the two version in some cases may not return the same value. In an attempt to move toward the solution of this issue we consider

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j \oplus \frac{1}{2}\big(\mathrm{app}(-\mathbf{u}_j \oplus \mathbf{u}) + \mathrm{app}(\mathbf{u} \oplus -\mathbf{u}_j)\big) \end{cases} \tag{4.5}$$

Writing directly the approximations and using the BCH approximation of degree 1 it become:

$$\begin{cases} \mathbf{u}_0 = 0 \\ \mathbf{u}_{j+1} = \mathbf{u}_j + \frac{1}{2}\big(\exp(-\mathbf{u}_j)\exp(\mathbf{u}) - e + \exp(\mathbf{u})\exp(-\mathbf{u}_j) - e\big) \end{cases} \tag{4.6}$$

Experimental results of the methods presented in this section are presented in the next chapter.

Chapter 5

Experimental Results

"A victory is twice itself when the achiever brings home full numbers."
Much ado about nothing*, Leonato, scene 1.*

In **chapter 1** the concept of log-composition is introduced, with its implications in medical imaging, as in diffeomorphic registration and in the computation of the logarithm in the log-Euclidean framework. **Chapter 2** is devoted to the introduction of the underpinning mathematical theory. Here are presented three numerical methods for the computation of the log-composition:

1. Truncated BCH formula of degree $k = 1, 2, 3$ - equation 2.10.

2. Taylor expansion - equation 2.11.

3. Parallel transport - equation 2.12.

To evaluate their performance, two groups of transformations are presented in **chapter 3** with respective numerical methods for their log-composition:

1. The finite dimensional Lie group of euclidean transformation SE(2) - section 3.1

2. The infinite dimensional Lie group diffeomorphisms, set of images of SVF through the Lie exponential map - section 3.2

The computation of the Lie logarithm as an important piece in the jigsaw puzzle of the log-euclidean framework is presented in **chapter 4** within the log-algorithm [BO08]. The numerical methods tailored for the computation of the log-algorithm here proposed are

1. Truncated BCH formula of degree $k = 1, 2, 3$ - equation 4.2.

2. Parallel transport - equation 4.4.

3. Symmetric parallel transport - equation 4.5.

In this last chapter we will compare the results of the numerical methods presented so far. The performance of the log-composition with each of the presented methods, applied to the groups of transformations $SE(2)$ and diffeomorphisms parametrized by SVF, is evaluated over synthetic datasets.

5.1 Log-composition for $\mathfrak{se}(2)$

5.1.1 Methods

A set of 500 transformations in $\mathfrak{se}(2)$ are sampled with a randomizer. Whereas the Frobenius norm of the matrix representation of $SE(2)$ is not proportional to the rotation angle θ, in $\mathfrak{se}(2)$ it is

$$||(\theta, dt_x, dt_y)||_{\text{fro}} = \sqrt{2\theta^2 + dt_x^2 + dt_y^2}$$

on with frobenius norm between 0.0 and 1.0

5.1.2 Results

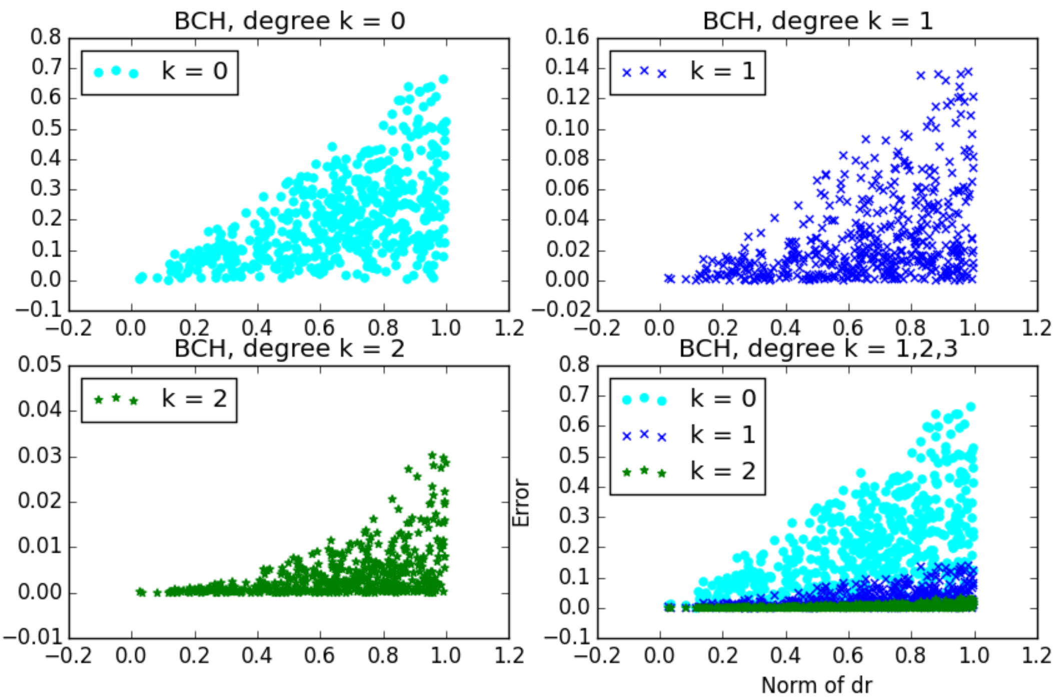

Figure 5.1: comparisons between BCH methods. TODO

5.1.3 Empirical Evaluations of Computational Time

5.2 Log-composition for SVF

Using a python software, we created synthetic random matrices in $\mathfrak{se}(2)$. We considered the difference between the log composition computed with the closed form 3.1. A sample of 500 couples (dr_0, dr_1) of elements in $\mathfrak{se}(2)$ are created,

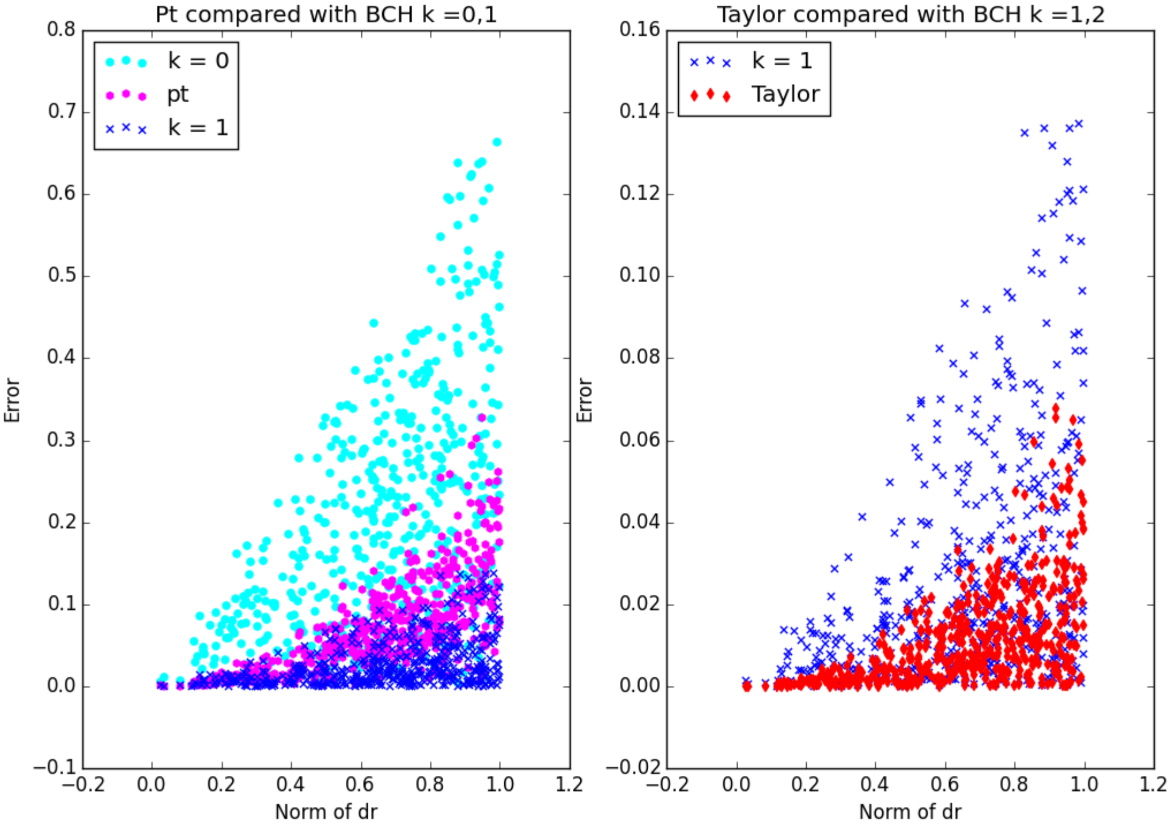

Figure 5.2: comparisons between Parallel Transport method and Taylor method. TODO

5.2.1 Methods

5.2.2 Results

5.2.3 Empirical Evaluations of Computational Time

5.3 Log-Algorithm for SVF

5.3.1 Methods

5.3.2 Results

5.3.3 Empirical Evaluations of Computational Time

5.4 Conclusion and Further Research

Considering only the results, this one-year research can be considered much ado about nothing, but...
Computational time...!

Starting from the definition of Lie log-group of diffeomorpshisms $(\mathfrak{g}, \oplus)$, to have an algebraic definition of this approximation, we can consider its quotient over the ideal generated by $(\mathrm{ad}^m_{\mathbf{u}}, \mathrm{ad}^n_{\mathbf{u}})$, which provides the group $(\mathfrak{g}/(\mathrm{ad}^m_{\mathbf{u}}, \mathrm{ad}^n_{\mathbf{u}}), \oplus)$. Further investigations in this

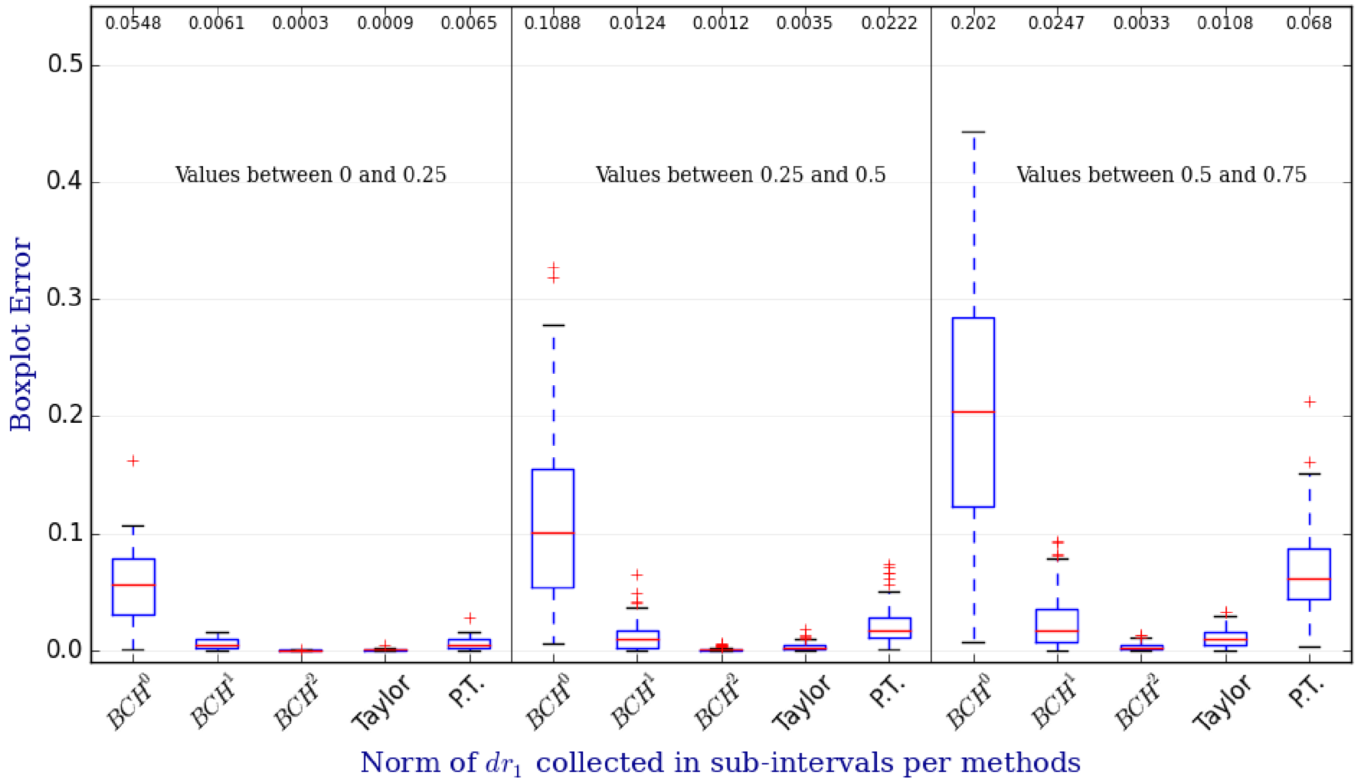

Figure 5.3: comparisons between all the methods. TODO

direction is not prosecuted.

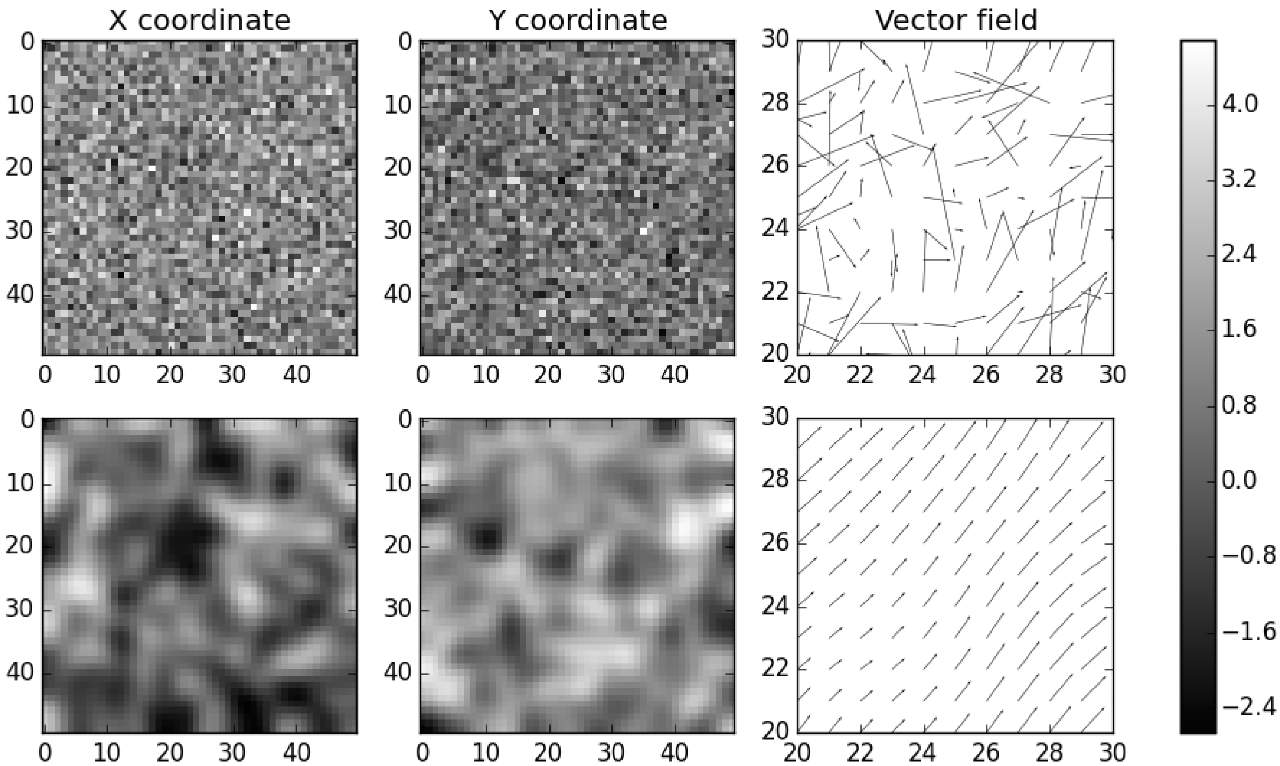

Figure 5.4: Random generated vector field:

Bibliography

[ACPA06a] Vincent Arsigny, Olivier Commowick, Xavier Pennec, and Nicholas Ayache. A log-euclidean framework for statistics on diffeomorphisms. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*, pages 924–931. Springer, 2006.

[ACPA06b] Vincent Arsigny, Olivier Commowick, Xavier Pennec, and Nicholas Ayache. Statistics on diffeomorphisms in a log-euclidean framework. In *1st MICCAI Workshop on Mathematical Foundations of Computational Anatomy: Geometrical, Statistical and Registration Methods for Modeling Biological Shape Variability*, pages 14–15, 2006.

[AFPA06] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine*, 56(2):411–421, August 2006.

[APA06] Vincent Arsigny, Xavier Pennec, and Nicholas Ayache. Bi-invariant means in lie groups. application to left-invariant polyaffine transformations. 2006.

[Arn66] Vladimir Arnold. Sur la géométrie différentielle des groupes de lie de dimension infinie et ses applications à l'hydrodynamique des fluides parfaits. In *Annales de l'institut Fourier*, volume 16, pages 319–361. Institut Fourier, 1966.

[Arn98] Vladimir Igorevich Arnold. *Topological methods in hydrodynamics*, volume 125. Springer Science & Business Media, 1998.

[Arn06] Vladimir I Arnold. Ordinary differential equations. translated from the russian by roger cooke. second printing of the 1992 edition. universitext, 2006.

[Art11] Michael Artin. Algebra. 2nd, 2011.

[Ash07] J. Ashburner. A fast diffeomorphic image registration algorithm. *NeuroImage*, 38(1):95–113, 2007.

[BBHM11] MARTIN BAUER, MARTINS BRUVERIS, PHILIPP HARMS, and PETER W MICHOR. Geodesic distance for right invariant sobolev metrics of fractional order on the diffeomorphism group. 2011.

[BHM10] Martin Bauer, Philipp Harms, and Peter W Michor. Sobolev metrics on shape space of surfaces in n-space. *Arxiv preprint arXiv:1009.3616*, 2010.

[BMTY05] M 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.

[BO08] Matias Bossa and Salvador Olmos. A New Algorithm for the Computation of the Group Logarithm of Diffeomorphisms. In Xavier Pennec and Sarang Joshi, editors, *Second International Workshop on Mathematical Foundations of*

Computational Anatomy - Geometrical and Statistical Methods for Modelling Biological Shape Variability, New York, USA, 2008.

[Bro81] Chaim Broit. *Optimal Registration of Deformed Images.* PhD thesis, Philadelphia, PA, USA, 1981. AAI8207933.

[CBD+03] Pascal Cachier, Eric Bardinet, Didier Dormont, Xavier Pennec, and Nicholas Ayache. Iconic feature based nonrigid registration: the pasha algorithm. *Computer vision and image understanding*, 89(2):272–298, 2003.

[CRM96] Gary E Christensen, Richard D Rabbitt, and Michael I Miller. Deformable templates using large deformation kinematics. *Image Processing, IEEE Transactions on*, 5(10):1435–1447, 1996.

[DCDC76] Manfredo Perdigao Do Carmo and Manfredo Perdigao Do Carmo. *Differential geometry of curves and surfaces*, volume 2. Prentice-hall Englewood Cliffs, 1976.

[dCV92] Manfredo Perdigao do Carmo Valero. *Riemannian geometry.* 1992.

[DGM98] Paul Dupuis, Ulf Grenander, and Michael I. Miller. Variational problems on flows of diffeomorphisms for image matching, 1998.

[EM70] David G Ebin and Jerrold Marsden. Groups of diffeomorphisms and the motion of an incompressible fluid. *Annals of Mathematics*, pages 102–163, 1970.

[EMP06] DAVID G Ebin, GERARD Misiołek, and STEPHEN C Preston. Singularities of the exponential map on the volume-preserving diffeomorphism group. *Geometric and Functional Analysis*, 16(4):850–868, 2006.

[FF97] Nick C Fox and Peter A Freeborough. Brain atrophy progression measured from registered serial mri: validation and application to alzheimer's disease. *Journal of Magnetic Resonance Imaging*, 7(6):1069–1075, 1997.

[Gal11] Jean Gallier. *Geometric methods and applications: for computer science and engineering*, volume 38. Springer Science & Business Media, 2011.

[GM98] Ulf Grenander and Michael I Miller. Computational anatomy: An emerging discipline. *Quarterly of applied mathematics*, 56(4):617–694, 1998.

[Gra88] Janusz Grabowski. Free subgroups of diffeomorphism groups. *Fundamenta Mathematicae*, 2(131):103–121, 1988.

[GWRNJ12] Serge Gauthier, Liyong Wu, Pedro Rosa-Neto, and Jianping Jia. Prevention strategies for alzheimer's disease. *Translational neurodegeneration*, 1(1):1–4, 2012.

[Hal15] Brian Hall. *Lie groups, Lie algebras, and representations: an elementary introduction*, volume 222. Springer, 2015.

[HBO07] Monica Hernandez, Matias N Bossa, and Salvador Olmos. Registration of anatomical images using geodesic paths of diffeomorphisms parameterized with stationary vector fields. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[HOP08] Monica Hernandez, Salvador Olmos, and Xavier Pennec. Comparing algorithms for diffeomorphic registration: Stationary lddmm and diffeomorphic demons. In *2nd MICCAI workshop on mathematical foundations of computational anatomy*, pages 24–35, 2008.

[HS81] Berthold K Horn and Brian G Schunck. Determining optical flow. In *1981 Technical Symposium East*, pages 319–331. International Society for Optics and Photonics, 1981.

[KKK08] Alexander A Kirillov, Aleksandr Aleksandrovič Kirillov, and Aleksandr Aleksandrovič Kirillov. *An introduction to Lie groups and Lie algebras*, volume 113. Cambridge University Press Cambridge, 2008.

[KMN00] Arkady Kheyfets, Warner A Miller, and Gregory A Newton. Schild's ladder parallel transport procedure for an arbitrary connection. *International Journal of Theoretical Physics*, 39(12):2891–2898, 2000.

[Kne51] MS Knebelman. Spaces of relative parallelism. *Annals of Mathematics*, pages 387–399, 1951.

[KO89] S Klarsfeld and JA Oteo. The baker-campbell-hausdorff formula and the convergence of the magnus expansion. *Journal of physics A: mathematical and general*, 22(21):4565, 1989.

[KW08] Boris Khesin and Robert Wendt. *The geometry of infinite-dimensional groups*, volume 51. Springer Science & Business Media, 2008.

[LAF+13] Marco Lorenzi, Nicholas Ayache, Giovanni B Frisoni, Xavier Pennec, and Alzheimer's Disease Neuroimaging Initiative. LCC-Demons: a robust and accurate symmetric diffeomorphic registration algorithm. *NeuroImage*, 81:470–483, 2013.

[LAP11] Marco Lorenzi, Nicholas Ayache, and Xavier Pennec. Schild's ladder for the parallel transport of deformations in time series of images. In *Information Processing in Medical Imaging*, pages 463–474. Springer, 2011.

[Lee12] John Lee. *Introduction to smooth manifolds*, volume 218. Springer Science & Business Media, 2012.

[Les83] JA Leslie. A lie group structure for the group of analytic diffeomorphisms. *Boll. Un. Mat. Ital. A (6)*, 2:29–37, 1983.

[LP13] Marco Lorenzi and Xavier Pennec. Geodesics, parallel transport & one-parameter subgroups for diffeomorphic image registration. *International journal of computer vision*, 105(2):111–127, 2013.

[LP14a] Marco Lorenzi and Xavier Pennec. Discrete Ladders for Parallel Transport in Transformation Groups with an Affine Connection Structure. In Frank Nielsen, editor, *Geometric Theory of Information*, Signals and Communication Technology, pages 243–271. Springer, 2014.

[LP14b] Marco Lorenzi and Xavier Pennec. Efficient parallel transport of deformations in time series of images: from schild's to pole ladder. *Journal of Mathematical Imaging and Vision*, 50(1-2):5–17, 2014.

[MA70] J Marsden and R Abraham. Hamiltonian mechanics on lie groups and hydrodynamics. *Global Analysis,(eds. SS Chern and S. Smale), Proc. Sympos. Pure Math*, 16:237–244, 1970.

[MHM+11] JR McClelland, S Hughes, M Modat, A Qureshi, S Ahmad, DB Landau, S Ourselin, and DJ Hawkes. Inter-fraction variations in respiratory motion models. *Physics in medicine and biology*, 56(1):251, 2011.

[MHSK] J. R. McClelland, D. J. Hawkes, T. Schaeffter, and A. P. King. Respiratory motion models: A review. *Medical Image Analysis*, 17(1):19–42, 2015/04/01.

[Mil82] John Milnor. On infinite dimensional lie groups. *Preprint, Institute for Advanced Study, Princeton*, 1982.

[Mil84a] J Milnor. Remarks on infinite-dimensional lie groups, in 'relativity, groups and topology, ii'(les houches, 1983), 1007–1057, 1984.

[Mil84b] John Milnor. Remarks on infinite-dimensional lie groups. In *Relativity, groups and topology. 2*. 1984.

[MT13] Carlo Mariconda and Alberto Tonolo. *Calcolo discreto: Metodi per contare*. Apogeo Editore, 2013.

[MTW73] Charles W Misner, Kip S Thorne, and John Archibald Wheeler. *Gravitation*. Macmillan, 1973.

[OKC92] V Yu Ovsienko, BA Khesin, and Yu V Chekanov. Integrals of the euler equations of multidimensional hydrodynamics and superconductivity. *Journal of Soviet Mathematics*, 59(5):1096–1101, 1992.

[Omo70] Hideki Omori. On the group of diffeomorphisms on a compact manifold. In *Proc. Symp. Pure Appl. Math., XV, Amer. Math. Soc*, pages 167–183, 1970.

[PCA99] Xavier Pennec, Pascal Cachier, and Nicholas Ayache. Understanding the "demon's algorithm": 3d non-rigid registration by gradient descent. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI'99*, pages 597–605. Springer, 1999.

[PCL+15] Ferran Prados, Manuel Jorge Cardoso, Kelvin K Leung, David M Cash, Marc Modat, Nick C Fox, Claudia AM Wheeler-Kingshott, Sebastien Ourselin, Alzheimer's Disease Neuroimaging Initiative, et al. Measuring brain atrophy with a generalized formulation of the boundary shift integral. *Neurobiology of aging*, 36:S81–S90, 2015.

[PL+11] Xavier Pennec, Marco Lorenzi, et al. Which parallel transport for the statistical analysis of longitudinal deformations. In *Proc. Colloque GRETSI*. Citeseer, 2011.

[Sch10] Rudolf Schmid. Infinite-dimensional lie groups and algebras in mathematical physics. *Advances in Mathematical Physics*, 2010, 2010.

[SDP13] A. Sotiras, C. Davatzikos, and N. Paragios. Deformable medical image registration: A survey. *Medical Imaging, IEEE Transactions on*, 32(7):1153–1190, July 2013.

[Ser09] Jean-Pierre Serre. *Lie algebras and Lie groups: 1964 lectures given at Harvard University*. Springer, 2009.

[SS09] Elias M Stein and Rami Shakarchi. *Real analysis: measure theory, integration, and Hilbert spaces*. Princeton University Press, 2009.

[Sze94] Richard Szeliski. Image mosaicing for tele-reality applications. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pages 44–53. IEEE, 1994.

[Thi98] J-P Thirion. Image matching as a diffusion process: an analogy with maxwell's demons. *Medical image analysis*, 2(3):243–260, 1998.

[Tro98] Alain Trouvé. Diffeomorphisms groups and pattern matching in image analysis. *International Journal of Computer Vision*, 28(3):213–221, 1998.

[Ver14] Tom Vercauteren. Technical memo. Pre-print, 2014.

[VPM+06] Tom Vercauteren, Aymeric Perchant, Grégoire Malandain, Xavier Pennec, and Nicholas Ayache. Robust mosaicing with correction of motion distortions and tissue deformations for in vivo fibered microscopy. *Medical image analysis*, 10(5):673–692, 2006.

[VPPA07] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Non-parametric diffeomorphic image registration with the demons algorithm. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007*, pages 319–326. Springer, 2007.

[VPPA08] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Symmetric log-domain diffeomorphic registration: A demons-based approach. In Dimitris N. Metaxas, Leon Axel, Gabor Fichtinger, and Gábor Székely, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2008, 11th International Conference, New York, NY, USA, September 6-10, 2008, Proceedings, Part I*, volume 5241 of *Lecture Notes in Computer Science*, pages 754–761. Springer, 2008.

[VRRC12] François-Xavier Vialard, Laurent Risser, Daniel Rueckert, and Colin J Cotter. Diffeomorphic 3d image registration via geodesic shooting using an efficient adjoint calculation. *International Journal of Computer Vision*, 97(2):229–241, 2012.

[War13] Frank W Warner. *Foundations of differentiable manifolds and Lie groups*, volume 94. Springer Science & Business Media, 2013.

[Woj94] Wojciech Wojtyński. One-parameter subgroups and the bch formula. *Studia Mathematica*, 111(2):163–185, 1994.

[You10] Laurent Younes. Shapes and diffeomorphisms, volume 171 of applied mathematical sciences, 2010.

[ZF03] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and vision computing*, 21(11):977–1000, 2003.