



HAL
open science

Geodesic methods and learning for ultrasound localization microscopy imaging

Théo Bertrand

► **To cite this version:**

Théo Bertrand. Geodesic methods and learning for ultrasound localization microscopy imaging. Mathematics [math]. Université Paris sciences et lettres, 2024. English. NNT: 2024UPSLD024 . tel-04873550

HAL Id: tel-04873550

<https://theses.hal.science/tel-04873550v1>

Submitted on 8 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à l'Université Paris-Dauphine

**Méthodes Géodésiques et Apprentissage pour le Microscopie
par Localisation Ultrasonore**

Soutenue par

Théo BERTRAND

Le 20 septembre 2024

École doctorale n°543

Ecole Doctorale SDOSE

Spécialité

Mathématiques appliquées

Composition du jury :

| | |
|--|---------------------------|
| Jean-Marie MIREBEAU Directeur de recherche CNRS, ENS Paris-Saclay | <i>Président du Jury</i> |
| Abderrahim ELMOATAZ Professeur des Universités, Université de Caen Basse Normandie | <i>Rapporteur</i> |
| Pierre WEISS Directeur de Recherche CNRS, IRIT | <i>Rapporteur</i> |
| Vincent DUVAL Directeur de Recherche, INRIA | <i>Examineur</i> |
| Olivier COUTURE Directeur de Recherche, Sorbonne Université, CNRS, INSERM | <i>Examineur</i> |
| Fang YANG Maîtresse de conférence, Wuhan University of Science and Technol- ogy | <i>Examinatrice</i> |
| Laurent COHEN Directeur de recherche CNRS, Université Paris-Dauphine PSL | <i>Directeur de thèse</i> |

Université Paris-Dauphine PSL

Doctoral School **École Doctorale Sciences de la Décision, des Organisations, de la
Société et de l'Échange**

University Department **Centre De Recherche en Mathématiques de la Décision,
Université Paris-Dauphine, UMR CNRS 7534**

Thesis defended by **Théo Bertrand**

Defended on **September 20, 2024**

In order to become Doctor from Université Paris-Dauphine PSL

Academic Field **Sciences**

Speciality **Applied Mathematics**

Méthodes géodésiques et Apprentissage pour la Microscopie par Localisation Ultrasonore

Thesis supervised by Laurent D. COHEN

Committee members

| | | | |
|-------------------|---------------------|---|---------------------|
| <i>Referees</i> | Abderrahim ELMOATAZ | Professor at Université de Caen Normandie | |
| | Pierre WEISS | Senior Researcher at IRIT | |
| <i>Examiners</i> | Jean-Marie MIREBEAU | Senior Researcher at ENS Paris-Saclay | Committee President |
| | Vincent DUVAL | Senior Researcher at INRIA | |
| | Olivier COUTURE | Senior Researcher at Sorbonne Université | |
| | Fang YANG | Associate Professor at Wuhan University of Science and Technology | |
| <i>Supervisor</i> | Laurent D. COHEN | Senior Researcher at Université Paris Dauphine-PSL | |

COLOPHON

Doctoral dissertation entitled “Méthodes géodésiques et Apprentissage pour la Microscopie par Localisation Ultrasonore”, written by Théo BERTRAND, completed on November 14, 2024, typeset with the document preparation system \LaTeX and the `yathesis` class dedicated to theses prepared in France.

Université Paris-Dauphine PSL

École doctorale **École Doctorale Sciences de la Décision, des Organisations, de la
Société et de l'Échange**

Unité de recherche **Centre De Recherche en Mathématiques de la Décision,
Université Paris-Dauphine, UMR CNRS 7534**

Thèse présentée par **Théo Bertrand**

Soutenue le **20 septembre 2024**

En vue de l'obtention du grade de docteur de l'Université Paris-Dauphine PSL

Discipline **Sciences**

Spécialité **Mathématiques Appliquées**

Geodesic and Learning methods for Ultrasound Localization Microscopy

Thèse dirigée par Laurent D. COHEN


Composition du jury

| | | | |
|---------------------------|---------------------|--|-------------------|
| <i>Rapporteurs</i> | Abderrahim ELMOATAZ | professeur à l'Université de Caen Normandie | |
| | Pierre WEISS | directeur de recherche à l'IRIT | |
| <i>Examineurs</i> | Jean-Marie MIREBEAU | directeur de recherche à l'ENS Paris-Saclay | président du jury |
| | Vincent DUVAL | directeur de recherche à l'INRIA | |
| | Olivier COUTURE | directeur de recherche au Sorbonne Université | |
| | Fang YANG | MCF au Wuhan University of Science and Technology | |
| <i>Directeur de thèse</i> | Laurent D. COHEN | directeur de recherche à l'Université Paris Dauphine-PSL | |

This thesis has been prepared at

**Centre De Recherche en Mathématiques
de la Décision, Université Paris-Dauphine,
UMR CNRS 7534**

Place du Maréchal De Lattre De Tassigny
75016 Paris
France

 +33 1 44 27 42 98

Web Site <http://ceremade.dauphine.fr>

CEREMADE
UMR CNRS 7534

A ma famille.

Mankind, be vigilant... I loved
you.

Julius Fučík

Méthodes géodésiques et Apprentissage pour la Microscopie par Localisation Ultrasonore**Abstract**

La méthode de Microscopie par Localisation Ultrasonore (ULM) est une nouvelle méthode d'imagerie médicale super-résolue qui permet d'outrepasser le compromis précision/distance de pénétration dans les tissus pour l'imagerie du réseau vasculaire. Ce nouveau type d'images pose de nouveaux enjeux mathématiques, notamment pour la segmentation et l'analyse de ses images, étapes nécessaires pour arriver à des méthodes pour le diagnostic. Notre travail se positionne à l'intersection des méthodes géodésiques et des méthodes de Machine Learning. Dans cette thèse nous apportons trois contributions. Une première de ces contributions est centrée autour des contraintes liées aux images ULM et propose le tracking de l'entièreté de l'arbre vasculaire en passant par la détection des point-clés des vaisseaux sanguins apparaissant sur l'image. La deuxième contribution de cette thèse porte sur l'apprentissage de la définition de métriques Riemanniennes pour traiter des tâches de segmentation sur des données d'IRM cérébraux et d'images du fond de l'oeil. La dernière partie de notre travail porte sur un problème inverse pour la reconstruction de trajectoires d'agents de contraste dans des images médicales dans le contexte de la super-résolution sans-grille.

Keywords: Vision par ordinateur, Distances géodésiques, Courbes géodésiques, Problèmes inverses, Machine learning

Geodesic and Learning methods for Ultrasound Localization Microscopy**Résumé**

Ultrasound Localization Microscopy is a new method in super-resolved Medical Imaging that allow us to overcome compromise between precision and penetration distance in the tissues for the imaging of the vascular network. This new type of images raises new mathematical questions, especially for the segmentation and analysis, necessary steps to achieve medical diagnostic of patients. Our work is positioned at the intersection of geodesic and Machine Learning methods. In this thesis, we make three contributions. The first of these is centered on the constraints linked to ULM images and proposes the tracking of the entire vascular tree through the detection of key points of blood vessels appearing on the image. The second contribution of this thesis deals with learning to define Riemannian metrics to handle segmentation tasks on brain MRI data and eye fundus images. The final part of our work focuses on an inverse problem for reconstructing contrast agent trajectories in medical images in the context of grid-free super-resolution.

Mots clés : Computer Vision, Geodesic distances, Geodesic curves, Inverse Problems, Machine learning

Acknowledgements

La rédaction de cette thèse fut longue et difficile, ainsi il est nécessaire pour moi d'exprimer quelques remerciements aux personnes qui ont aidé de près ou de loin à cette entreprise.

Tout d'abord j'aimerais remercier les membres du jury pour le temps consacré à la lecture de ce manuscrit, son commentaire et leur participation à la soutenance de la thèse.

Un grand merci également à mon directeur de thèse Laurent Cohen pour son encadrement scientifique, ses conseils et son soutien matériel pour mener à bien mes recherches.

Le quotidien de la thèse aurait également été beaucoup moins supportable sans la présence d'autres doctorants et mathématiciens sur-doués qui ont pu devenir des amis. Ainsi j'étends mes remerciements à Lorenzo mon co-bureau à la culture et aux talents culinaires extraordinaires, à Charly mon co-bureau mélomane, bavard et toujours prêt à m'expliquer des trucs, à Bastien mon co-auteur et ami depuis Centrale avec qui j'ai souffert en parallèle, à Antoine Lotz mon ami depuis la prépa également, à Antoine Luciano le plus grand cool kid zoomer que j'ai rencontré jusque là, à Emma la cool kid et ma partenaire de piscine, à Daniele il pazzo scalatore di Parma, à Umberto le plus fidèle soldat de Bologne la Rouge, à João mon canard préféré, à Adrien, imbattable en pronostics politiques et compagnon de pauses, à Ryad le Goldman boy du CEREMADE, à Donato le sage italien, à Rodrigue (Qu'est-ce que t'es belle !), à Jean, à Quan, à Giovanni, à Camilla, à Katharina, à Grégoire, à Florin, à Richard, et à tous les autres.

Je n'aurais pas non plus pu garder ma santé mentale sans mes amis de longue date. Ceux qui me suivent depuis la prépa : Freddy, Théo, Eli, Thomas, Anne-Laure, Isabelle, Guilhem, Louis, Léo. Ainsi que Guillaume (et Ambre !) et Léa. A ceux qui me soutiennent depuis le lycée : Elisa, Raphaëlle, Marin, Justine. A mes camarades de Centrale : Gael et Bob (que j'ai trop peu vus ces dernières années), Eloi, Anaïs, Antoine M. et tout le reste du V3. Et il y en a certainement des tas que j'oublie.

Un grand merci aussi aux permanents qui m'ont aidé pour les enseignements, le soutien moral ou la gestion du labo : Emeric, Katia, Matthieu et Vincent. Un merci infini également aux personnels du secrétariat et de l'informatique qui savent nous rendre la vie tellement plus facile : Isabelle, Anne-Laure, César, Marko, Thomas et Gilles.

Merci évidemment à mes parents et ma soeur qui m'ont soutenu pendant la thèse mais également pendant toutes les années d'étude qui ont précédé.

Ces travaux ont été financés par l'Agence Nationale de la Recherche, rela-

tivement au programme "Investissements d'avenir", référence ANR-19-P3IA-0001 (PRAIRIE 3IA Institute).

Contents

| | |
|--|-------------|
| Abstract | xiii |
| Acknowledgements | xv |
| Contents | 1 |
| Résumé en Français | 3 |
| Schéma et résumé des contributions | 3 |
| La Microscopie par Localisation Ultrasonore | 5 |
| Tracking complet de l'arbre vasculaire par détection et classification de points-clé sur les images de fond d'oeil et ULM | 5 |
| Apprentissage de métrique Riemannienne | 8 |
| Recouvrement de courbes par un problème variationnel dynamique, extension aux espaces de position et d'orientation | 9 |
| Reproductibilité | 12 |
| 1 Introduction | 13 |
| 1.1 Outline and Summary of the contributions | 13 |
| 1.2 Ultrasound Localization microscopy | 15 |
| 1.2.1 Acquisition | 16 |
| 1.2.2 Localization | 16 |
| 1.2.3 Tracking | 17 |
| 1.2.4 Mathematicians in ULM : What is to be done ? | 19 |
| 2 Geodesic distances and curves | 21 |
| 2.1 Definitions and properties | 21 |
| 2.1.1 A very short introduction to Riemannian geometry | 27 |
| 2.2 The Eikonal equation | 29 |
| 2.2.1 The Fast Marching Algorithm | 30 |
| 2.2.2 Heat Method | 35 |
| 2.2.3 Other methods | 37 |
| 2.3 Geodesic Methods in Image Processing | 40 |
| 3 Machine Learning for Computer Vision | 43 |
| 3.1 Machine Learning, Deep Learning : First Principles | 43 |
| 3.1.1 Supervised Learning | 43 |
| 3.1.2 Optimization : automatic differentiation and gradient descent | 45 |

| | | |
|----------|---|------------|
| 3.1.3 | Neural Network architectures | 46 |
| 3.2 | Tracking vascular trees on medical images | 50 |
| 3.2.1 | Introduction | 50 |
| 3.2.2 | Detecting vascular landmarks | 53 |
| 3.2.3 | Finding appropriate geodesics | 55 |
| 3.2.4 | Results and Discussion | 61 |
| 3.2.5 | Evaluation on segmentation score and discussion | 68 |
| 3.2.6 | Partial Conclusion | 70 |
| 3.3 | Riemannian metric learning | 74 |
| 3.3.1 | Model | 76 |
| 3.3.2 | Experiments | 78 |
| 3.3.3 | Segmentation Experiments | 80 |
| 3.3.4 | More examples | 82 |
| 3.3.5 | Segmenting blood vessels by learning anisotropy | 89 |
| 3.3.6 | Partial Conclusion | 91 |
| 4 | Analysis on the space of paths and Optimal Transport | 93 |
| 4.1 | Introduction to Optimal Transport | 93 |
| 4.1.1 | The Monge and Kantorovitch problem | 94 |
| 4.1.2 | The Dynamic Formulation | 96 |
| 4.1.3 | Optimal Transport as measures on the space of paths | 97 |
| 4.2 | Benamou-Brenier regularization for off-the-grid Curve recovery in the space of positions and orientations | 100 |
| 4.2.1 | Introduction and contributions | 100 |
| 4.2.2 | An energy for trajectory recovery | 102 |
| 4.2.3 | The orientation-position space | 104 |
| 4.2.4 | Discretisation of the space of curves, beyond polygonal curves | 108 |
| 4.2.5 | Numerical Results | 112 |
| 4.2.6 | Partial Conclusion and outlook | 121 |
| 4.2.7 | Appendix : Extreme points of the Benamou-Brenier Energy (Riemannian case) | 122 |
| 4.2.8 | Appendix : Computations for the relaxed Reeds-Shepp | 123 |
| 4.2.9 | Appendix : Riemannian optimization on Manifold and Tangent bundle | 124 |
| 4.2.10 | Appendix : Numerical results in the detail | 126 |
| | Conclusion and Future Works | 131 |
| | Bibliography | 133 |

Résumé en Français

Les travaux présentés dans cette thèse visent à fournir de nouvelles méthodes numériques pour l'analyse d'images médicales, la segmentation de structures vasculaires et d'autres organes, avec un intérêt particulier pour les applications aux récentes techniques d'imagerie par microscopie par localisation ultrasonore. Nous fournissons un formalisme mathématique pour nos résultats théoriques et nos heuristiques. Le but de ce manuscrit est d'être relativement autonome et de permettre aux lecteurs ayant des connaissances de niveau M2 en mathématiques et en méthodes numériques de comprendre les contributions à l'état de l'art présentées ici.

Ce travail s'inscrit dans le contexte des méthodes géodésiques utilisées pour aborder les tâches de vision par ordinateur, et tente de combiner les deux mondes des modèles mathématiques issus du calcul des variations et des approches modernes d'apprentissage automatique/apprentissage en profondeur.

Schéma et résumé des contributions

Cette thèse est séparée en 4 chapitres thématiques. Tout d'abord, l'introduction se concentre sur une présentation générale de la thèse et introduit les méthodes de microscopie de localisation par ultrasons et les problèmes qui y sont liés. Ensuite, le deuxième chapitre vise à présenter les fondements théoriques des méthodes géodésiques et leurs pendants numériques, nous voulons également donner une idée générale des méthodes existantes dans le domaine. Notre troisième chapitre est une tentative d'introduction rapide et générale au domaine très vaste des méthodes d'apprentissage automatique en vision par ordinateur, suivie de la présentation de nos deux premières contributions à l'état de l'art en matière de vision par ordinateur pour lequel il était nécessaire d'introduire la théorie de la distance géodésique au chapitre 2 et l'apprentissage automatique au début du chapitre 3. Le chapitre 4 est peut-être le plus indépendant puisqu'il n'a que peu de liens avec les autres chapitres, mais il introduit rapidement le Transport Optimal afin de pouvoir présenter la dernière contribution de notre travail, à savoir une étude de l'extension d'un modèle récent proposé pour la récupération de courbes à partir d'une suite d'acquisitions.

Cette thèse a été l'occasion de produire trois travaux différents publiés ou soumis à revue par les pairs, qui couvrent différentes parties du domaine.

Notre premier travail original est présenté en détail dans la section 3.2, il s'agit d'une première contribution permettant de réaliser le tracking de vaisseaux sanguins dans des images ULM dans le contexte particulier où il n'y a pas d'annotation de

données pour une tâche de segmentation, et de tirer parti autant que possible de l'information fournie par les données ULM dans ce contexte. Ce travail est également l'occasion de tester certaines des hypothèses sous-jacentes à l'état de l'art des méthodes de tracking des vaisseaux sanguins. En effet, beaucoup de publications se concentrent sur la définition de bons modèles pour le tracking de structures curvilignes dans des images naturelles en 2D ou 3D, et fournissent souvent de bonnes méthodes pour des problèmes bien définis et relativement simples tels que la recherche de géodésiques entre deux points pour effectuer le tracking, mais tendent à oublier que le tracking n'est en pratique qu'une partie d'un modèle plus large. Nous essayons ici de présenter une heuristique explicite pour notre modèle de bout en bout pour le tracking de l'ensemble de l'arbre vasculaire. Il s'agit d'un réseau neuronal entraîné à proposer des points de repère de l'arbre vasculaire, on sélectionne des sous-arbres minimisant la longueur via la définition d'une distance géodésique dépendant de l'orientation locale des caractéristiques. Notre workflow est présenté sur 3 ensembles de données différents, dont un composé de très peu d'images ULM. À la connaissance des auteurs, il s'agit du premier travail dans le domaine des méthodes géodésiques à tirer parti des informations ULM pour définir une telle géométrie.

Notre deuxième travail original est présenté en détail dans la section 3.3, il montre qu'il est possible d'intégrer la distance géodésique et les informations qui en découlent directement dans un pipeline d'apprentissage automatique. Notre objectif est de pouvoir générer une géométrie directement à partir des données de l'image sans que l'utilisateur ait à choisir un modèle. Nous avons utilisé une architecture CNN classique pour apprendre et généraliser la définition d'une métrique isotrope pour la segmentation des tumeurs cérébrales, segmentation que nous retrouvons en prenant une approximation lisse de l'indicatrice de la boule unité pour distance définie par la métrique. La différentiation de la distance géodésique par rapport au potentiel est rendue possible par le calcul d'un sous-gradient à l'aide de l'algorithme de Subgradient Marching (et a été comparée à d'autres méthodes d'approximation de la distance géodésique). Elle montre des performances similaires à celles obtenues dans l'état de l'art avec des garanties théoriques supplémentaires sur le masque de segmentation fourni.

Le troisième et dernier travail original présenté dans ce manuscrit est détaillé dans la section 4.2. On traite ici des problèmes de recouvrement des déplacements d'objets dans une pile d'acquisition. En particulier, on est intéressé par le recouvrement de la trajectoire d'agents de contraste dans les méthodes d'imagerie médicale par super-résolution, dont la Microscopie par Localisation Ultrasonore, mais aussi plus largement pour le recouvrement de données de trajectoire (données GPS, déplacements de cellules,...). Pour la super-résolution, on veut notamment pouvoir récupérer les trajectoires elles-mêmes, en particulier en ayant de bonnes informations sur les champs de vitesse qui peuvent permettre (comme on le voit dans notre premier travail) des applications intéressantes. Les données sont constituées d'une pile comportant à chaque temps la réponse d'un opérateur linéaire évalué sur les objets à retrouver, représentés par un ensemble continu de mesures indexées par le temps. Il s'appuie sur un modèle récent proposé pour récupérer les trajectoires sous-jacentes d'objets en mouvement dans une pile d'acquisition. Il s'agit d'un problème inverse

défini dans l'espace des mesures sur les courbes continues. Nos contributions incluent la proposition d'étendre ce cadre à la géométrie non-euclidienne pour récupérer avec précision les courbes de croisement en 2D en élevant le problème à l'espace des positions et des orientations, et en fournissant également un cadre garantissant la $(\Gamma-)$ convergence du problème discrétisé vers le problème continu et la proposition d'autres espaces de discrétisation dans ce cadre. La convergence est également valable pour des termes d'attache aux données calculés sur des espaces de dimension infinie, ce qui est plus général que le cas traité dans la littérature qui s'appuie sur le théorème du représentant. Nous évaluons notre modèle en exécutant un simple algorithme de Frank-Wolfe, en résolvant le problème d'optimisation non convexe à chaque étape à l'aide de la différentiation automatique, et nous comparons les cas euclidien et position-orientation, ainsi que les espaces de discrétisation.

La Microscopie par Localisation Ultrasonore

La Microscopie par Localisation Ultrasonore (ULM) est une méthode d'imagerie moderne utilisée pour produire des images du réseau vasculaire. Bien qu'elle utilise des ondes ultrasonores pour réaliser l'imagerie, elle permet d'obtenir des images très précises des vaisseaux sanguins, jusqu'à quelques microns. En effet, les techniques classiques d'imagerie par diffusion inverse des ondes souffrent d'un inconvénient important : elles ne peuvent pas pénétrer trop profondément dans les tissus sans que le signal ne devienne trop faible. Il faut donc trouver un compromis entre l'imagerie de haute précision, qui est possible en utilisant des signaux de plus haute fréquence, et l'atténuation du signal et la distance de pénétration correspondante, qui diminuent à mesure que les fréquences augmentent.

L'imagerie par Microscopie par Localisation Ultrasonore contourne ce compromis en introduisant des *agents de contraste* spéciaux dans le sujet à observer sous la forme de microbulles de gaz inerte. Ces microbulles peuvent être localisées avec précision en séparant la réponse non linéaire des bulles par rapport à la réponse linéaire des tissus. Après la localisation des microbulles, on peut reconstruire une image en projetant les positions sur une grille et en formant un histogramme 2D ou 3D de la position des microbulles. La résolution des images ULM dépend donc du nombre de bulles détectées. Elle fait partie d'un ensemble de techniques d'imagerie appelées techniques de "super-résolution", car elles permettent, dans un certain sens, de dépasser les limites théoriques des systèmes d'imagerie.

Nous présentons maintenant un peu plus précisément nos trois apports à l'état de l'art :

Tracking complet de l'arbre vasculaire par détection et classification de points-clé sur les images de fond d'oeil et ULM

Nous introduisons une nouvelle méthode pour la détection complète de bout en bout des structures vasculaires sur les images ULM, en utilisant l'apprentissage profond pour détecter les points caractéristiques permettant le suivi des vaisseaux en tant qu'arêtes dans un arbre (au sens des graphes) avec des points-clés en tant que sommets. Notre approche diffère du Regroupement perceptuel classique pour

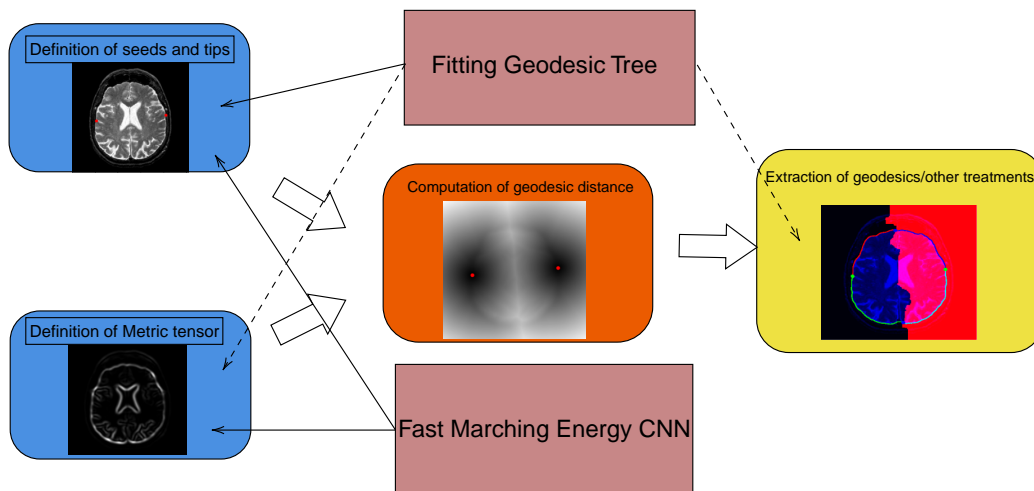


Figure 1: Schéma stéréotypé de la démarche des méthodes géodésiques. En rouge nos deux travaux originaux. Les flèches pleines indiquent les étapes revisitées à l'aide du Machine Learning et en pointillés les étapes revisitées avec un modèle "à la main".

le suivi des vaisseaux sanguins tel qu'il est effectué dans d'autres travaux E. J. Bekkers, D. Chen, et al. 2018. En effet, nous essayons de tirer profit des longues géodésiques qui suivent les vaisseaux sanguins à travers une image, ce qui devrait bien se comporter compte tenu de la quantité de littérature sur le sujet. Alors que le Regroupement perceptuel se concentre généralement sur le calcul de géodésiques courtes entre des points proches répartis dans le réseau de vaisseaux, notre objectif est de calculer quelques géodésiques longues entre des points de repère clés de la vascularisation. Nous essayons de tirer parti des informations fournies spécifiquement par l'imagerie ULM, mais il convient de noter qu'il est possible de s'adapter à d'autres types d'images, par exemple les images du fond de l'œil obtenues par photographie directe. Pour ce faire, il peut être nécessaire d'évaluer les informations d'orientation locale, comme nous le verrons dans la section 3. À partir d'une image 2D, on peut utiliser des scores d'orientation ou des transformations similaires telles que celles présentées dans E. Bekkers et al. 2014 pour relever une image 2D dans le plan à l'espace tridimensionnel des positions et des orientations.

L'imagerie ULM n'ayant pas de base de données comportant des images naturelles associées à une annotation de segmentation il est particulièrement pertinent d'adopter une approche de tracking des vaisseaux sanguins. En effet l'enjeu devient alors de définir les points particuliers de l'image à relier et un modèle adéquat pour le calcul de chemins minimaux. L'idée principale est que la structure sous-jacente à l'agencement des vaisseaux sanguins dans les images médicales est celle d'un arbre. Celui-ci est composé de vaisseaux sanguins agencés de manière à transporter de manière efficace les composés chimiques charriés par le sang. Cet agencement en arbre impose donc qu'il existe des terminaisons aux endroits où le sang est délivré ou aux endroits où les vaisseaux sortent de l'image (que ce soit sur les côtés, ou en profondeur pour les images 2D) ainsi que des bifurcations, zones où un vaisseau se

sépare en plusieurs vaisseaux. Dans les images 2D on peut également voir apparaître des vaisseaux qui se chevauchent dans des zones qu'on appelle alors croisements. Ces trois catégories de points remarquables dans le réseau vasculaire permettent en fait de définir sa topologie, le reste n'est plus alors qu'une question de savoir relier correctement les points entre eux. Comme la littérature en méthodes géodésiques s'est beaucoup attachée à définir des modèles pour le tracking de vaisseaux sanguins, on a à notre disposition des modèles pour relier deux points en passant par les vaisseaux appropriés. Le réseau vasculaire étant supposé efficace du point de vue de l'irrigation des tissus et des organes (c'est-à-dire qu'il permet d'irriguer les zones à irriguer tout en restant aussi court que possible), on va essayer de le retrouver en cherchant les arbres passant par les point-clefs et minimisant la longueur totale pour la notion de distance que l'on aura choisie. Cette heuristique est au coeur de ce qui se fait historiquement en méthodes géodésiques et de la façon dont on conçoit les réseaux vasculaires. Ce travail permet d'interroger ces présupposés qui sont souvent utilisés pour penser les modèles de tracking des vaisseaux sanguins.

On va appliquer notre approche sur trois différents jeux de données. Le premier est simplement un lot de deux images dessinées à la main qui nous ont permis de faire un premier essai en imitant la configuration de données ULM : on a deux images disponibles de haute résolution. Le second jeu de données est un ensemble de deux grandes images synthétiques, dessinées à la main pour simuler un cas très simple et semblable aux données ULM disponibles. Pour ces deux jeux de données on a annoté à la main les terminaisons, les bifurcations et les croisements sur ces images. Notre motivation étant qu'il est en principe plus facile pour un praticien non-expert de détecter ces structures que de générer une annotation de segmentation complète. Notre troisième jeu de données est un jeu de données classique pour la segmentation, pour lequel les auteurs de Abbasi-Sureshjani et al. 2015 ont mis à disposition une annotation des point-clés du réseau vasculaire.

La parcimonie des données dans le cas synthétique et le cas ULM est contournée en découpant de petits morceaux de l'image de manière uniforme afin de former une base de données plus fournie. L'augmentation de données par application de transformations affines aléatoires aide également beaucoup à éviter le surentraînement.

La sortie du UNet entraîné comporte 4 canaux, un pour chaque classe de point à trouver et un dernier qui représente le max des autres canaux, faisant ainsi qu'il est plus intéressant de détecter un point même en prédisant la mauvaise classe plutôt que de ne rien prédire du tout.

L'entraînement dure quelques heures. Les résultats de la détection des point-clés sont très convaincants comme attendu sur les données synthétiques avec des scores atteints autour de 80%. Sur les données du fond de l'oeil, les scores atteints sont autour des 40% sur la base de test. Ces résultats sont légèrement décevants par rapport à ceux annoncés dans la littérature adoptant une approche similaire. On remarque cependant que c'est la catégorie des terminaisons de vaisseaux sanguins qui semblent les plus difficiles à détecter, celles-ci n'étant pas détectées dans Hervella et al. 2019.

Les résultats sur données ULM en revanche n'atteignent qu'environ 20% de score F1. Il est à noter que l'annotation des données ULM n'est certainement pas parfaite et permet d'expliquer en partie ces mauvais résultats : il est difficile d'annoter

de manière systématique les structures intéressantes alors qu'il y a énormément d'échelles différentes représentées dans une même image ULM.

Le tracking de vaisseaux sanguins se fait ensuite en utilisant un modèle de Reeds-Shepp relaxé permettant de décroiser les vaisseaux en posant le problème de chemins minimaux dans un espace de dimension supérieure : on ajoute aux coordonnées de position une coordonnée correspondant à l'orientation locale. Ce modèle permet de trouver des vaisseaux sanguins qui se croisent mais ayant deux orientations différentes, les géodésiques ne doivent en principe pas circuler de l'un à l'autre. Ce modèle de Reeds-Shepp comporte une fonction de coût en facteur qui est définie à partir des niveaux de gris de l'image. Pour les données ULM, ce score a été construit à partir des données ULM de trajectoires en formant une image comme l'histogramme dans l'espace de position-orientation, en prenant comme orientation l'estimation de vitesse de chaque particule disponible au sein des trajectoires. Pour les données synthétiques et du fond l'oeil, on applique des filtres classiques de relèvement d'images en niveaux de gris, ici en convoluant avec un noyau gaussien anisotrope orienté dans N_θ directions différentes.

Une difficulté de notre approche est qu'il n'est pas évident d'évaluer les performances du tracking, en tout cas de fournir une base de comparaison pertinente avec les méthodes de segmentation.

Apprentissage de métrique Riemannienne

Les courbes et distances géodésiques ont été utilisées pour transmettre des propriétés géométriques dans de nombreuses applications différentes. L'approche présentée dans ce travail tente de se débarrasser du biais introduit dans le choix d'un tenseur métrique en le générant à partir de données via une architecture de réseau neuronal dont les paramètres ont été préalablement optimisés dans une approche d'apprentissage supervisé avec des données d'apprentissage. L'introduction d'un tel biais n'est pas une mauvaise chose en soi, mais elle nécessite une décision arbitraire de la part de l'utilisateur et un réglage des paramètres, deux problèmes qui peuvent être évités en apprenant à générer une métrique à partir de données. En utilisant la méthode proposée, nous pouvons obtenir des résultats précis par rapport aux approches traditionnelles, ce qui met en évidence les capacités de cette approche. La méthode introduite dans ce travail offre un moyen puissant et flexible d'utiliser les courbes géodésiques et les distances dans une large gamme d'applications dans un cadre d'apprentissage holistique.

Notre approche a été de s'intéresser à un problème classique de segmentation de tumeurs dans des images de cerveaux, plus précisément une collection de sections de cerveaux sur lesquelles sont annotées les tumeurs. Ayant fixé comme objectif de traiter cette tâche classique, nous avons cherché à générer un masque de segmentation à l'aide d'une distance géodésique, en faisant l'hypothèse qu'un masque de segmentation pouvait être retrouvé comme la boule unité pour une distance géodésique associée à un potentiel et un centre bien choisis dans l'image. Pour respecter cette hypothèse on écarte toutes les images du jeu de données dont le masque de segmentation comporte plusieurs composantes connexes ainsi que celles où le masque est identiquement nul.

Afin de générer la métrique à partir de l'image on va utiliser une architecture de réseau de neurones classique en vision par ordinateur : le UNet. Celui-ci est légèrement modifié en lui ajoutant une seconde branche de décodeur. La première branche est utilisée pour générer le potentiel et la seconde pour générer une carte de probabilités 2D de présence d'un bon centre pour la boule géodésique. A partir de l'annotation de segmentation, le centre utilisé comme référence à apprendre est défini comme le barycentre de l'annotation, c'est une approximation efficace qui fonctionne car les zones à segmenter dans les images de tumeur sont assez proches d'être convexes et de ressembler à des boules déformées. L'apprentissage du potentiel est quasi-découplé en utilisant l'annotation du centre pour le calcul de la boule au moment de l'apprentissage, mais en évaluant la moyenne de la carte de probabilité générée par le modèle au moment de la validation et du test.

La fonction objectif optimisée durant l'entraînement est donc :

$$\mathbb{E}_{(x,y) \sim \mathcal{D}} \left[\mathcal{L}_S(f_\theta^1(x), y) + \mathcal{L}_B(f_\theta^2(x), k * \delta_{\text{Bar}(y)}) \right], \quad (1)$$

où f_θ^1 et f_θ^2 sont les sorties des deux branches décodeur du réseau paramétré par $\theta \in \mathbb{R}^p$, \mathcal{D} la distribution théorique des images d'entrée x et leur annotation associée y . k est simplement un noyau gaussien utilisé afin de donner une épaisseur à la position du barycentre et qu'elle soit ainsi plus facile à prédire. \mathcal{L}_S est une fonction de coût entre la segmentation prédite et la segmentation à trouver. \mathcal{L}_B est une fonction de coût entre le barycentre prédit et le barycentre théorique.

La fonction-coût pour la segmentation est composée d'un coût classique entre le masque proposé et le masque théorique (distance L^2 , terme d'entropie croisée) et d'une façon de transformer la carte de distances en masque pour la proposition, ici on utilisera une fonction sigmoïde afin de proposer une approximation lisse de la fonction indicatrice de l'ensemble $\{d(x_0, \cdot) \leq 1\}$.

L'implémentation a été faite en PyTorch et on retrouve des scores de segmentation proches de ce qui apparaît dans l'état de l'art. Les potentiels produits par notre architecture ressemblent aux sorties d'une méthode de détection de contours, stoppant la propagation du front avant que celui-ci ne sorte de la zone à segmenter lors du calcul de la distance. On montre également une réelle disparité parmi les résultats avec de très bons scores de segmentation sur une grosse majorité du jeu de données mais de très mauvaises performances sur une petite proportion, tirant ainsi le score moyen vers le bas.

On va ensuite essayer d'étendre ce cadre à la segmentation de régions et classes multiples, puis à l'apprentissage de métriques anisotropes.

Recouvrement de courbes par un problème variationnel dynamique, extension aux espaces de position et d'orientation

Nous considérerons un modèle de problème inverse similaire à celui proposé dans Bredies et Fanzon 2019 pour lequel Duval et Tovey 2021 ont proposé des méthodes numériques, tandis que Schmitzer et al. 2019 semble être le premier travail à proposer la régularisation considérée. L'idée principale est de résoudre un problème inverse en construisant une mesure supportée sur un espace de courbes minimisant

une fonctionnelle d'énergie avec un terme de données qui pénalise la distance entre les données et le minimiseur candidat, et un terme de régularisation qui est linéaire dans la mesure de minimisation candidate. Cette formulation permet à l'utilisateur de donner des garanties sur la forme des minimiseurs. En effet, ce problème a été dérivé en gardant à l'esprit l'opportunité d'utiliser des théorèmes des Représentants comme ceux prouvés dans Boyer et al. 2018 ; Bredies et Carioni 2018, car ils garantissent que la forme du minimiseur est déterminée par les points extrémaux de la boule unité du régularisateur. L'utilisation de ce théorème du représentant nécessite la satisfaction de quelques hypothèses, parmi lesquelles la dimensionnalité finie de l'espace sur lequel la distance aux données est calculée. Dans Bredies et Carioni 2018, les auteurs donnent une preuve que les points extrémaux des boules unitaires pour la régularisation considérée sont des mesures de Dirac sur l'espace des courbes, ce qui garantit la parcimonie de la solution.

La fonctionnelle suivante est l'élément central du cadre dynamique sans-grille Bredies, Carioni, Fanzon, et al. 2021; Bredies and Fanzon 2019 jusqu'à présent :

$$E(\sigma) = \sum_{i=1}^T \|A_i e_{t_i} \# \sigma - b_{t_i}\|_{\mathcal{H}}^2 + \int_{\Gamma} w(\gamma) d\sigma(\gamma), \quad (2)$$

avec

$$\sigma \in \mathcal{M}(\Gamma),$$

et

$$w(\gamma) = \int_0^1 |\dot{\gamma}(t)|^2 dt.$$

\mathcal{H} est un espace de Hilbert, e_t est la forme linéaire $e_t : \gamma \in \Gamma \mapsto \gamma(t)$, et les A_i sont des opérateurs linéaires continus.

Notre première idée a d'abord été de réussir à obtenir des jeux de courbes croisées dans le cas simple traité dans les publications de l'état de l'art : l'énergie dans l'espace euclidien comme définie dans l'équation précédente ne permet pas en théorie d'obtenir deux courbes croisées dans X comme sur la Figure 2.

Pour cela nous choisissons de passer notre problème dans un espace de dimension supérieure, l'espace de Position-Orientation afin de pénaliser la courbure des trajectoires recouvertes.

$$\mathcal{P}_\varepsilon((x, \theta), (\dot{x}, \dot{\theta}))^2 = C((x, \theta))^2 (|\dot{x} \cdot e_\theta|^2 + \frac{1}{\varepsilon^2} |\dot{x} \wedge e_\theta|^2 + \xi^2 |\dot{\theta}|^2).$$

Après avoir vérifié que notre problème posé dans ce nouvel espace permet bien d'obtenir le terme de régularisation désiré, on vérifie également que comme dans le cas euclidien, les points extrémaux de la boule unité du nouveau terme de régularisation correspondent encore aux masses de Dirac sur l'espace des courbes. Ceci étant posé on sait qu'on pourra appliquer le théorème du représentant si \mathcal{H} est bien de dimension finie, c'est-à-dire qu'on sait alors que la solution au problème variationnel (2) peut s'écrire comme combinaison linéaire de points extrémaux du terme de régularisation, ou dans le cas où \mathcal{H} n'est pas de dimension finie, ce résultat est tout de même utile dans l'application de l'algorithme de Frank-Wolfe.

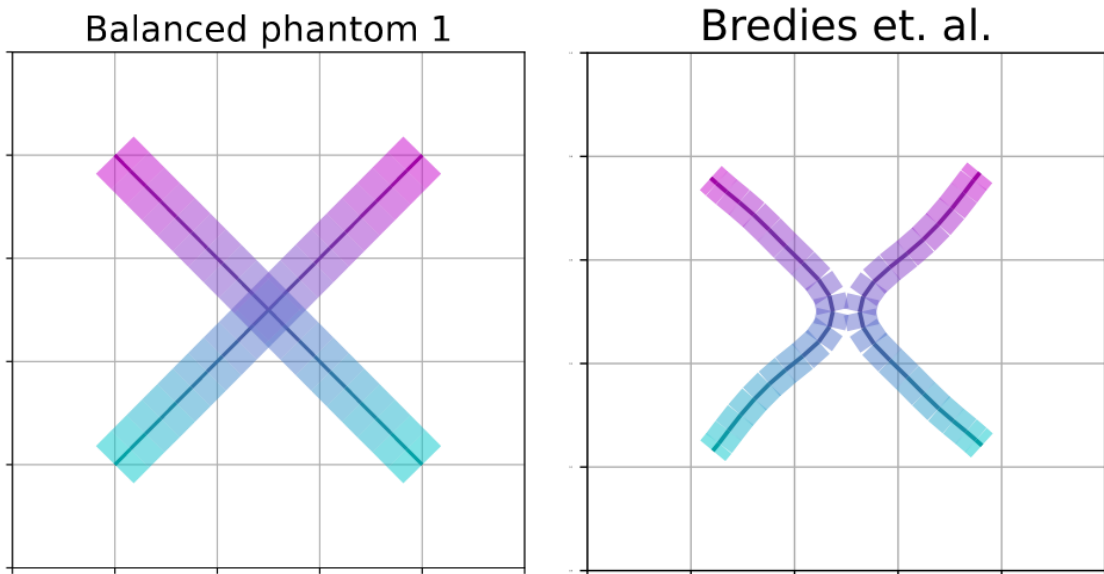


Figure 2: Reconstruction of two time dependent curves crossing from

Une autre partie de notre travail a été d'essayer d'élargir les options pour la discrétisation du problème. En effet, la littérature sur ce problème s'est en général concentrée sur la résolution numérique en cherchant des méthodes efficaces pour la résolution de l'oracle dans la méthode de Frank-Wolfe en cherchant toujours à approcher les trajectoires à retrouver par des courbes polygonales. Cette approche est justifiée par le fait que, pour un nombre fini de temps t_i dans le terme d'attache aux données, et étant donnée une mesure σ sur les courbes, modifier cette mesure en conservant les marginales aux temps t_i puis en interpolant les points du support entre les $e_{t_i\#}\sigma$ et $e_{t_{i+1}\#}\sigma$ par les courbes minimisant w permet de trouver un meilleur candidat.

Formellement on peut l'écrire comme ceci : soit

$$g : \gamma \mapsto \tilde{\gamma} \in \operatorname{argmin}\{w(h), h \in \Gamma, \forall i, h(t_i) = \gamma(t_i)\}, \quad (3)$$

une sélection mesurable de géodésiques entre deux dates (voir Duval and Tovey 2022, Lemme 3.3 pour l'existence), alors on a

$$E(g\#\sigma) \leq E(\sigma), \quad (4)$$

ce qui nous dit qu'en fait, la famille des courbes de géodésiques par morceaux interpolant entre les points définis à chaque date t_i est une classe optimale de courbes pour le problème continu et a fortiori pour le problème discret.

Notre critère pour déterminer qu'une discrétisation de l'espace des courbes est bonne sera de s'assurer que le problème posé dans l'espace des courbes discrétisées converge bien vers le problème continu lorsqu'on affine la discrétisation. On cherche donc à paramétriser les courbes par un ensemble de points de contrôle en définissant un opérateur $P^n : \mathbb{R}^{d \times k_n} \rightarrow \Gamma$ tel que chaque courbe γ puisse être approximée par une suite de paramètres. On définira également un opérateur de sélection $S_n : \Gamma \rightarrow \mathbb{R}^{d \times k_n}$ qui, à une courbe, associe des points de contrôle choisis de

manière à ce que

$$\forall \gamma \in \Gamma, P^n(S_n(\gamma)) \xrightarrow{n \rightarrow +\infty} \gamma. \quad (5)$$

Le premier terme dans l'énergie étant supposé continu, la Γ -convergence du problème discret vers le problème continu peut être garantie en imposant une propriété supplémentaire sur les opérateurs que nous venons de définir :

$$\forall \gamma \in \Gamma, \limsup_{n \rightarrow +\infty} w(P^n(S_n(\gamma))) \leq w(\gamma), \quad (6)$$

cette propriété est en fait à voir comme une relaxation de (4).

On vérifie aisément que les courbes géodésiques par morceaux vérifient

$$w(P^n(S_n(\gamma))) \leq w(\gamma) \quad (7)$$

dans le cas où $w = \mathcal{A}_2$ avec $\mathcal{A}_2(\gamma) = \int_0^1 |\dot{\gamma}|(t)^2 dt$, et les opérateurs P^n, S_n sont triviaux (c'est-à-dire en échantillonnant uniformément le long de la courbe), ce qui est évidemment une condition plus forte que (6).

Dans le cas euclidien on propose également les courbes de Bézier comme classe de courbes approximantes, qui vérifient également la dernière inégalité.

Reproductibilité

Tous les scripts et le code utilisés afin d'obtenir les résultats présentés dans ce manuscrit sont/seront disponibles librement à l'adresse <https://github.com/TheoBertrand-Dauphine> ou sont référencés via lien disponible via l'adresse précédente lorsque le code est hébergé par un coauteur par exemple.

Chapter 1

Introduction

The works presented in this thesis focus on providing new numerical methods for the analysis of medical images, segmentation of vascular structures and other organs, with a particular interest in the applications to the recent Ultrasound Localization Microscopy imaging techniques. We will provide mathematical formalism for our theoretical results and heuristics. The aim of this manuscript is to be somewhat self-contained and to allow readers with graduate-level understanding of mathematics and numerical methods to grasp the contributions to the state-of-the-art presented here.

This work fits in the context of Geodesic Methods used to tackle Computer Vision tasks, and attempts to combine mathematical and variational models and modern Machine/Deep Learning approaches.

1.1 Outline and Summary of the contributions

This thesis is thematically separated into 4 chapters. First, this Introduction focuses on a general presentation of the thesis and introduces the Ultrasound Localization Microscopy methods and related problems. Then, the second chapter aims at presenting the theoretical foundations of geodesic methods and their numerical counterparts, we also want to give a somewhat precise idea of the existing methods in the field. Our third chapter is an attempt at a quick and general introduction to the very wide field of Machine Learning methods in Computer Vision, followed by the presentation of our first two contributions to the state-of-the-art, for which it was necessary to first introduce the theory of Geodesic Distance in Chapter 2 and Machine Learning in the beginning of Chapter 3. Chapter 4 might be the most independent as it has only slight connections to Chapter 2, but it introduces quickly Optimal Transport in order to be able to present the last contribution of our work, namely a study of the extension of a recent model proposed for curve recovery.

As stated before, this thesis has been the opportunity to produce three different works published or submitted for review, which cover different parts of the domain.

Our first original work is presented in detail in Section 3.2, it carries out the tracking of blood vessels in ULM images in the peculiar context that there is no

data annotation for a segmentation task, and an attempt to take advantage as much as possible of the information provided by ULM data in this context. This work is also the opportunity to test some of the hypotheses underlying the state-of-the-art of the vessel tracking methods. Indeed, most publications in the field focus on the definition of good models for the tracking of curvilinear structures in 2D or 3D natural images, so they often provide good methods when looking at well-defined and relatively simple problems such as finding *good* geodesics between two points to carry out the tracking, but tend to forget that the tracking is in practice usually only some part of a larger workflow. Here we try to present an explicit heuristic for our end-to-end workflow for the tracking of the whole vascular tree. It involves a neural network trained to propose landmark points of the vascular tree, and we select the length-minimizing sub-trees via the definition of a geodesic distance depending on the local orientation of the features. Our workflow is presented on 3 different datasets, including one consisting of very few ULM images. To the knowledge of the authors it is the first work in the field of geodesic methods to take advantage of ULM information to define such a geometry.

Our second original work is presented in detail in Section 3.3. It shows that it is possible to integrate the geodesic distance and information derived from it directly in a machine learning pipeline. Our goal is to be able to generate a geometry directly from the image data without direct model choices from the user. We used a classic CNN architecture to learn and generalize the definition of an isotropic metric for the segmentation of brain tumours by matching the unit ball defined by the metric output by the network to the desired segmentation mask. Differentiation of the geodesic distance with respect to the potential is made available by the computation of a subgradient using the Subgradient Marching Algorithm (and was compared to other geodesic distance approximation methods). It shows performance similar to what is achieved in the state-of-the-art with additional theoretical guarantees on the segmentation mask provided.

The third and last of the original work presented in this manuscript is detailed in Section 4.2. We tackle the tracking of multiple objects moving in the plane as time evolves. The data consists in a pile of linear noisy data. This is typically useful in tracking contrast agents in super-resolution imaging methods, and in particular Ultrasound Localization microscopy, as well as general trajectory recovery tasks. For super-resolution imaging we want to recover trajectories and have access to accurate velocity information to be able to build models where for instance the orientation of the velocity is important (like in the first of our work). It builds on a recent model proposed to recover the underlying transient curves of moving objects in a pile of acquisition. It is an Inverse Problem defined on the space of measures on continuous curves. Our contributions include proposing to extend this framework to non-euclidean geometry to accurately recover crossing curves in 2D by lifting the problem to the space of positions and orientations, and also providing a framework guaranteeing the $(\Gamma-)$ convergence of the discretized problem to the continuous one and the proposal of other discretization spaces in that framework. The convergence is valid also for data terms computed on infinite dimensional spaces which is more general than the case treated in the literature which relies on the Representer Theorem. We evaluate our model by performing a simple Frank-Wolfe algorithm, solving

the non-convex optimization problem at each step using automatic differentiation, and compare the euclidean and position-orientation cases, and discretization spaces.

1.2 Ultrasound Localization microscopy

The goal of this subsection is to give a broad understanding of the Ultrasound Localization Microscopy imaging techniques and a brief overview of the methods involved in the formation of such images.

Ultrasound Localization Microscopy (ULM) is a modern imaging method that is used to produce images of the vascular network. Although it uses ultrasound waves to carry out the imaging, it allows for very precise images of the blood vessels, up to a few microns. Indeed classical imaging techniques via inverse scattering of waves suffer from an important drawback in that it cannot penetrate too deep in the tissues without the signal becoming too low. There's a trade-off to be made between imaging with high precision, which is possible by using signals with higher frequencies, and the *attenuation* of the signal and the related penetration distance, which diminishes as frequencies increase.

Ultrasound Localization Microscopy imaging circumvents this trade-off by introducing special *contrast agents* in the subject to be observed in the form of microbubbles of inert gas. Those microbubbles may be accurately localized by detecting the non-linear response of the bubbles versus the linear response of the tissues. After the localization of the microbubbles, one may reconstruct an image by projecting back the positions on a grid and forming a 2D or 3D histogram of the position of the microbubbles. The resolution of ULM images thus depends on the number of detected bubbles. It is part of a set of imaging techniques called "super-resolution" techniques, as they allow, in some sense, to overcome the theoretical limitations of the imaging systems.

This imaging technique shows many advantages among which are the ease of access to the tools. Indeed, ultrasound probes are relatively cheap and most hospitals are already equipped with them to carry out other types of interventions such as ultrasound scans for pregnancies, moreover, ultrasound probes are easy to use and do not necessitate highly trained practitioners to be used efficiently, as opposed to MRI imaging. It also has a few drawbacks, the main one being that it is intrusive as the injection of the contrast agent is needed, also the produced images depend on new parameters such as the concentration of the contrast agents and the time of measurement.

Couture, Besson, et al. 2011 and Siepmann et al. 2011, both from 2011, are the two foundational works for ULM imaging.

In the following, we will go a bit more in detail over the different steps of the process of forming ULM images. For thorough and complete overviews of this process one may refer to Dencks and Schmitz 2023; Chavignon 2021

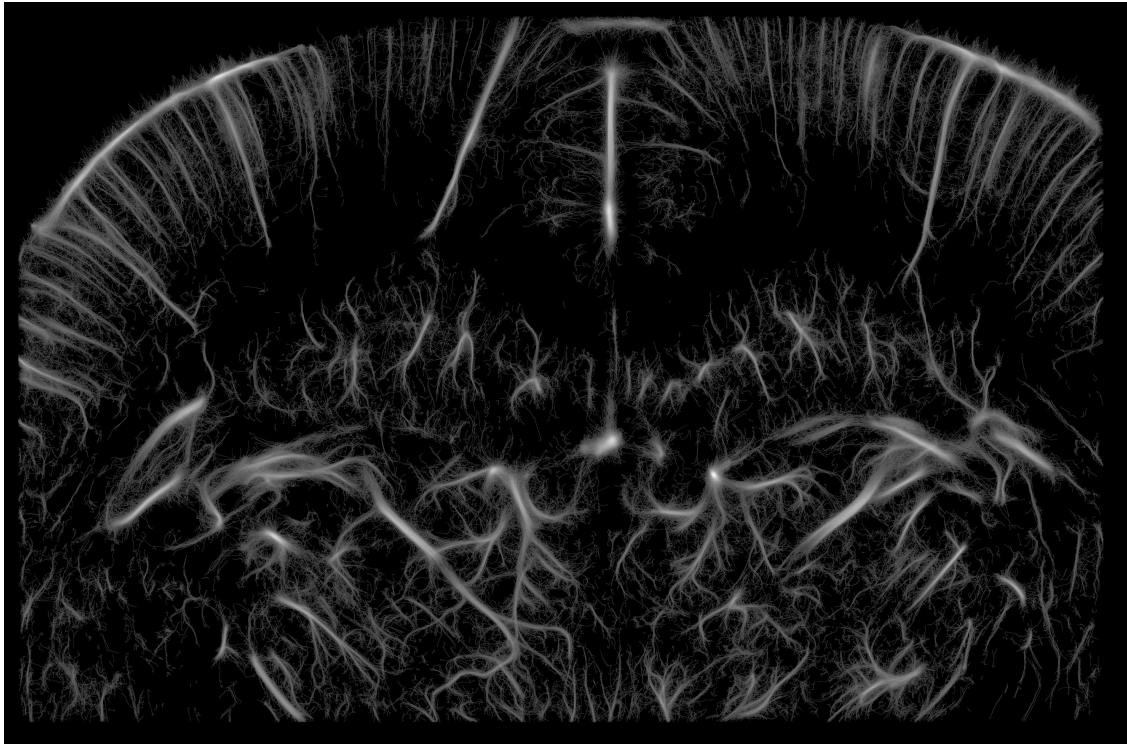


Figure 1.1: ULM image of a section of a rat brain. Data from Chavignon et al. 2020.

1.2.1 Acquisition

After the intraveinuous introduction of the contrast agents, the acquisition is performed using an ultrasound probe. The resulting signal consists in the sum of the responses of both the tissues and the microbubbles.

Extracting the response of the microbubbles from the response of the background tissues is a task that can be performed because we can separate the linear response of the tissue from the non-linear response of the microbubbles. Indeed, the microbubbles generally have harmonic resonances and responses in frequencies that don't match the frequency of the incoming signal. Usually, a relatively large bandwidth in receptor frequency is necessary (from $f_0/2$ to $2f_0$ for an initial frequency of f_0 , depending of the parameter of the process). Another approach is to make sums of signals with opposite phases, thus cancelling out the linear responses and leaving only the non-linear content.

1.2.2 Localization

Localization methods used in ULM imaging generally involve fitting a Point Spread Function in the neighbourhood of pixels with local maximum intensity. The authors of Heiles et al. 2022 (from which Figure 1.2 comparing different localization methods is reproduced here) have benchmarked the different algorithms used in the literature for localization of the microbubbles. The authors inspect numerous methods that have been used in the past to locate the microbubbles. Usually they consist in

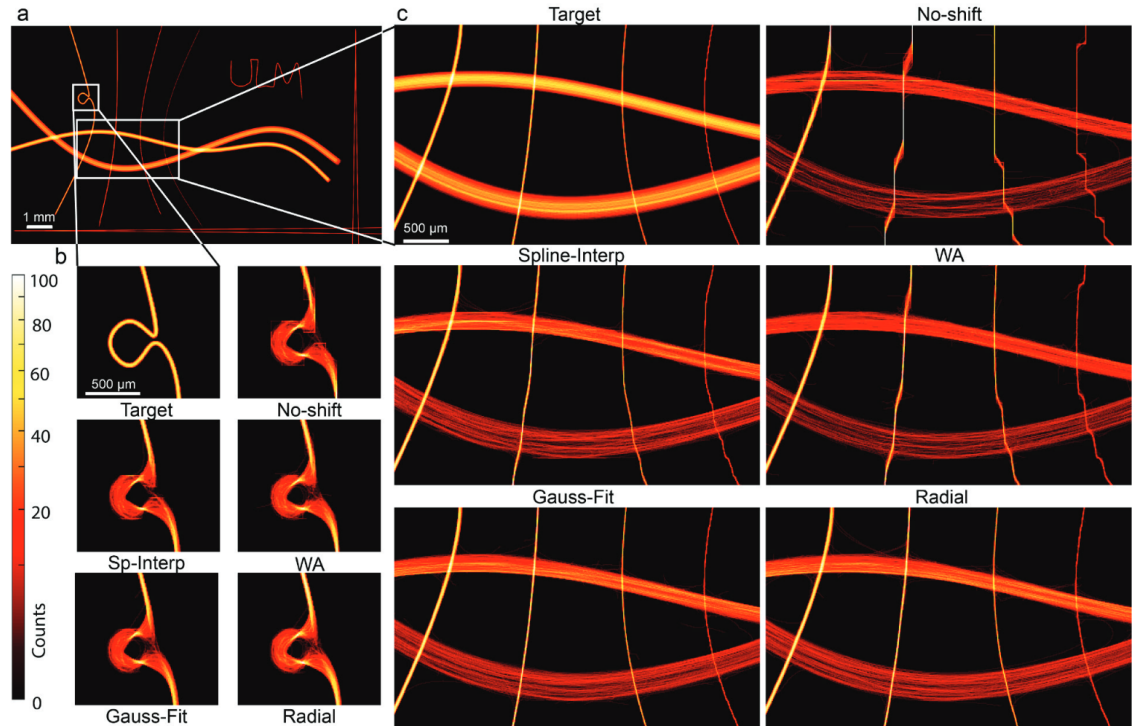


Figure 1.2: Comparison on in silico simulated flow of the performance of localization methods. Directly taken from Heiles et al. 2022

extracting a neighbourhood of a few pixels in size of local maxima in the grey-level image produced, and then estimating the center of the local distribution of amplitude in the patch.

The simplest and fastest method proposed is simply to compute the weighted average deviation from the center in the current patch. Heavier methods include interpolating the local distribution of amplitude with low order polynomials; also gaussian deconvolution, which consists in finding the best fitting gaussian kernel to approximate the local distribution in amplitude. The method which appear to show the best computation time/efficiency seem to be the "Radial Symmetry" algorithm which attempts to find the deviation from the central pixel in the patch by selecting the position minimizing the weighted average distance to the line defined by the gradient computed on the patch, see Parthasarathy 2012 for further details.

1.2.3 Tracking

Although direct formation of the image after localizing the microbubbles is possible, it is interesting to infer the trajectory of detected microbubbles to interpolate their positions between multiple frames and increase the number of points in the histogram.

Once the localization step is done and one has found a number of points in the image sequence, one can stop here and form an image like those in Figure 1.1 and Figure 1.4, but one can see also that we may infer the trajectory of the microbubbles we have detected even at timestamps that are not directly available to us. Indeed,

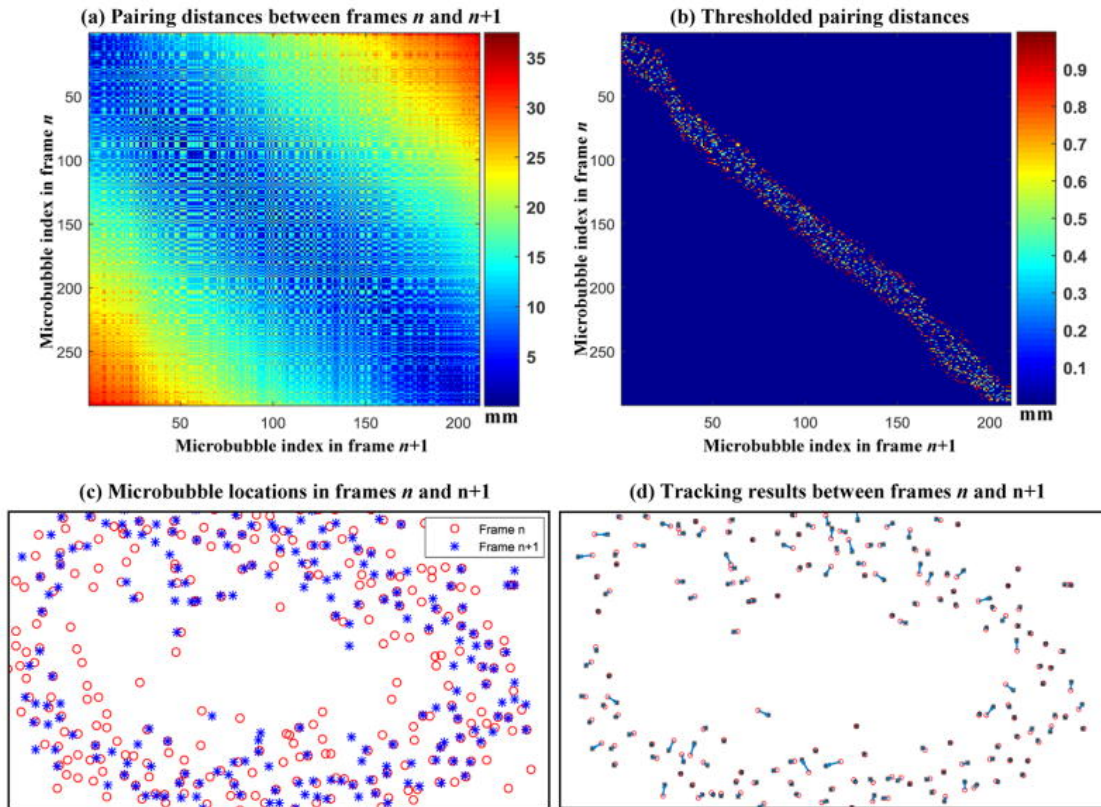


Figure 1.3: Illustration of the application of the Hungarian algorithm to the tracking of microbubbles. Taken from Song et al. 2018.

due to the time of survival of microbubbles in the blood, we know that it is very likely that a detected microbubble detected in a first shot would still be alive in the consecutive picture. Associating detected microbubbles from one acquisition to the next, one can infer the displacement of those microbubbles from one image of the sequence to the next. This way one can artificially enhance the total number of detected events and thus reduce the number of microbubbles to be detected in order to get a satisfying resolution of the image.

The tracking may be done using the Hungarian algorithm. This algorithm is usually used to find the best allocation of resources, meaning in mathematical terms that we will be looking for the best assignment on a bipartite graph. The way that is applied to Ultrasound Localization Microscopy is by considering the weighted bipartite graph of distances of detected microbubbles between two subsequent frames of the acquisition. One solves the matching problem on every consecutive pair of images and the two-by-two association of microbubbles defines chains that may be interpreted as trajectories. Those trajectories allow us to estimate velocity fields and flow rates.

Figure 1.3 shows the application of this method to the imaging of a rat's kidney (full picture shown in Figure 1.4).

The interpolation is then done by defining where the microbubble should be between two frames, usually it is defined by a straight line joining each pair of consecutive positions in the trajectory.

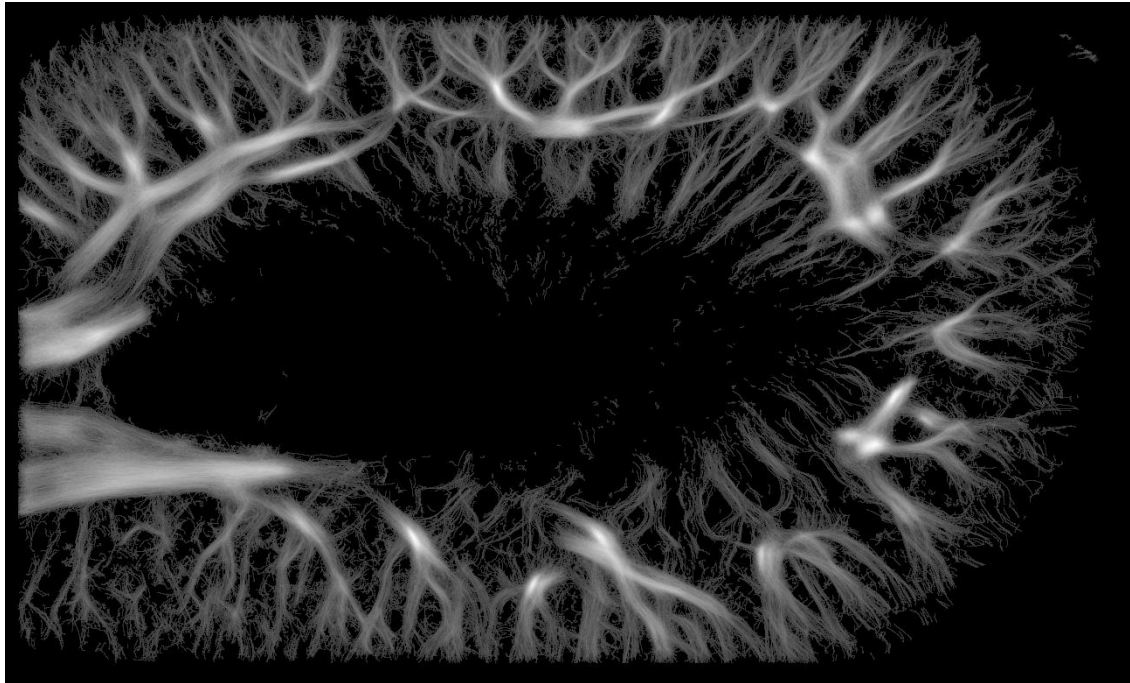


Figure 1.4: ULM image of a section of a rat's kidney. Data from Chavignon et al. 2020.

1.2.4 Mathematicians in ULM : What is to be done ?

There are still few works from applied mathematicians and computer scientist focusing in particular on ULM imaging. For instance, the authors in Pham et al. 2023 tried to improve the tissue filtering step by including a more sophisticated PCA in the ULM pipeline.

The literature seems to mainly focus on improving the imaging process with a few applications of deep learning aimed at specific steps of the process : for instance X. Chen, Lowerison, Dong, A. Han, et al. 2022; Youn, Ommen, Bo Stuart, et al. 2019; Harput et al. 2019; Youn, Ommen, Stuart, et al. 2020 focus on reducing the localisation error, on the other hand X. Chen, Lowerison, Dong, Chandra Sekaran, et al. 2023 is interested in recovering a velocity map directly from raw ultrasound data and without a localization step.

In Leconte et al. 2023 the authors show an interesting approach where the tracking is essentially performed before the localization, by using a classical tubularity filter on ULM data seen as three-dimensional $2D+t$ data.

This is approximately all there is to know in terms of the mathematics involved in the analysis of ULM images. As one can see, the field is still green and there is huge room for the development of new and original methods. One of the first things for the field to develop would be the availability of well annotated datasets for interesting and/or standard tasks on ULM images. Although the authors Chavignon et al. 2020 made the first step in this direction, the volume of available data is very important in the modern world, and the state-of-the-art machine learning techniques will not be accessible for ULM if practitioners don't play their role in the ecosystem. Although the lack of data is problematic, there still are steps to be

taken by engaging with machine learning that don't necessitate annotation of data such as unsupervised techniques in Machine Learning. On a more mathematical side, it looks like approaches directly involving trajectory data are rather underdeveloped and only a few papers focus on this type of data, usually for describing the movements of individuals or cells, but there seems to be, at least to our knowledge, no imaging technique or image analysis method focusing on using such data to describe medical images.

Geodesic distances and curves

The following definitions and properties are a minimum for the understanding of geodesic distances and curves for the analysis of images.¹

2.1 Definitions and properties

The determination of the *geodesic equation* is one of the most famous illustrations of Calculus of Variations and the application of Euler-Lagrange Equations. This variational framework is the one we are really interested in, although it may be necessary to go back to the formalism of Riemannian geometry.

First of all we introduce the geodesic distance and related optimization problems, first taking the very general point of view of Finsler metrics.

In the following, \mathcal{M} is at least a smooth path-connected manifold.

Definition 2.1 (Length and Minimal paths). A minimal path curve is a curve $\gamma \in \text{Lip}([0, 1], \mathcal{M})$ minimizing a length functional of the form

$$\mathcal{L}_{\mathcal{F}}(\gamma) = \int_0^1 \mathcal{F}(\gamma(t), \gamma'(t)) dt, \quad (2.1)$$

with constraints $\gamma(0) = x$ and $\gamma(1) = y$ for chosen $x, y \in \mathcal{M}$, and with $\mathcal{F} \geq 0$ continuous in the first variable and positively one-homogeneous in the second.

We may define the sub-Riemannian geodesic "distance" associated to \mathcal{F} by considering the minimum of the former objective functional :

$$d_{\mathcal{F}}(x, y) = \inf_{\gamma \in \text{Lip}([0, 1], \mathcal{M}), \gamma(0)=x, \gamma(1)=y} \mathcal{L}_{\mathcal{F}}(\gamma), \quad (2.2)$$

To proceed and prove some of the properties of the former objects and characterize the case when it defines a distance, we need the following lemma.

¹Curves and geodesics are fundamental ideas in the theory of Riemannian Geometry. Although we will sometimes need to refer to the formalism of Riemannian geometry and Calculus of Variation, when defining geodesic models we will generally use simple analytical expressions as it is often the case that we don't need to define standard objects from the theory of Riemannian geometry such as atlases, charts or connections. In general, applying geodesic models will only consist in considering a change from the euclidean metric tensor on \mathbb{R}^d as a sub-manifold of itself.

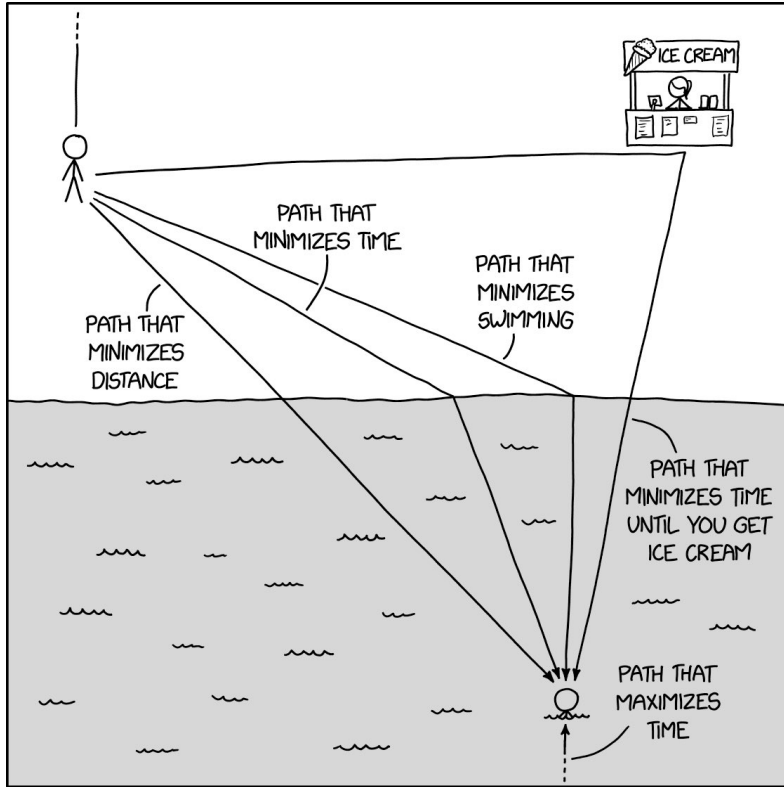


Figure 2.1: Illustration of minimal paths optimizing for different objectives. Source : <https://xkcd.com/>.

Lemma 2.2. $\mathcal{L}_{\mathcal{F}}$ is characterized by

$$\mathcal{L}_{\mathcal{F}}(\gamma) = \sup_{t_0=0 < t_1 < \dots < t_n=1} \sum_{k=0}^{n-1} d_{\mathcal{F}}(\gamma(t_k), \gamma(t_{k+1})),$$

with the supremum taken over the set of all finite subdivisions of the interval $[0, 1]$ and $\mathcal{L}_{\mathcal{F}}$ is a lower semicontinuous functional and parameterization independent.

Proof. Let's write $L_{\mathcal{F}}(\gamma) = \sup_{t_0=0 < t_1 < \dots < t_n=1} \sum_{k=0}^{n-1} d_{\mathcal{F}}(\gamma(t_k), \gamma(t_{k+1}))$, first it is pretty straightforward to notice that, for any subdivision $(t_k)_{0 \leq k \leq n}$, $\sum_{k=0}^{n-1} d_{\mathcal{F}}(\gamma(t_k), \gamma(t_{k+1})) \leq \int_0^1 |\gamma'(t)| dt$, thus yielding $L_{\mathcal{F}}(\gamma) \leq \mathcal{L}_{\mathcal{F}}(\gamma)$.

The inequality on the other side is proven by noticing that

$$d(\gamma(0), \gamma(1)) \leq L_{\mathcal{F}}(\gamma) \leq \mathcal{L}_{\mathcal{F}}(\gamma),$$

and that $s \mapsto L_{\mathcal{F}}(\gamma|_{[a,s]})$ is absolutely continuous and also

$$L_{\mathcal{F}}(\gamma|_{[a,s+h]}) \geq L_{\mathcal{F}}(\gamma|_{[a,s]}) + d(\gamma(s), \gamma(s+h))$$

which yields

$$\frac{d}{ds} L_{\mathcal{F}}(\gamma|_{[0,s]}) \geq |\gamma'(s)|,$$

which means

$$L_{\mathcal{F}}(\gamma) \geq \mathcal{L}_{\mathcal{F}}(\gamma).$$

And the supremum of the continuous maps $\gamma \mapsto \sum_i^{n-1} d(\gamma(t_i), \gamma(t_{i+1}))$ taken over all subdivisions of the interval is lower semicontinuous, see Ambrosio, Brué, et al. 2021.

For parameterization independence, it is easy to see using any reparametrization ϕ , an increasing, bijective, \mathcal{C}^1 function from $[0, 1]$ to $[0, 1]$, the equality $\mathcal{L}_{\mathcal{F}}(\gamma \circ \phi) = \mathcal{L}_{\mathcal{F}}(\gamma)$ follows from change of variable and 1-homogeneity of the integrand. \square

Note that the definition of parametrization invariance used in the former proposition is only considering reparametrization which conserve the direction of travel of the original curve. For symmetric metrics one may consider decreasing reparametrizations.

Proposition 2.3. *If $\mathcal{F}(x, \cdot)$ is definite, then $d_{\mathcal{F}}$ is a quasimetric (i.e. a distance without the symmetry axiom). Furthermore, if $\mathcal{F}(x, \cdot)$ is symmetric (or reversible) then $d_{\mathcal{F}}$ is a proper distance.*

Proof. • Positivity of $d_{\mathcal{F}}$ is obvious from the positivity of \mathcal{F}

- Separability : let γ^n be a minimizing sequence of curves, allowing $d_{\mathcal{F}}(x, y) = 0$ we have that for almost every $t \in [0, 1]$, $\lim_n \mathcal{F}(\gamma^n(t), \gamma'^n(t)) = 0$. Let us notice that definiteness of \mathcal{F} implies that $\forall(x, v) \in T\mathcal{M}, \beta\|v\| \leq \mathcal{F}(x, v)$, thus $\lim_n \|\gamma'^n(t)\| = 0$ and with dominated convergence theorem we easily get that $x = y$.
- For the triangular inequality, let $x, y, z \in \mathcal{M}$, and take γ_1^n, γ_2^n minimizing sequences of curves between respectively x and z , and z and y . Then we build $\tilde{\gamma}^n$ by concatenation of γ_1^n and γ_2^n , which means $\tilde{\gamma}^n$ is an admissible curve in (2.2), and by minimality we obtain $d_{\mathcal{F}}(x, y) \leq \int_0^1 \mathcal{F}(\tilde{\gamma}^n(t), \tilde{\gamma}'^n(t))dt$. Then separating the integral and by change of variable we obtain

$$d_{\mathcal{F}}(x, y) \leq \int_0^1 \mathcal{F}(\tilde{\gamma}_1^n(t), \tilde{\gamma}_1'^n(t))dt + \int_0^1 \mathcal{F}(\gamma_2^n(t), \gamma_2'^n(t))dt$$

and sending n to infinity we have that

$$d_g(x, y) \leq d_g(x, z) + d_g(z, y).$$

- Symmetry in the case where \mathcal{F} is reversible: Let γ be an optimal path, returning it in time we define $\tilde{\gamma}(t) = \gamma(1 - t)$, then

$$\begin{aligned} \int_0^1 \mathcal{F}(\tilde{\gamma}(t), \tilde{\gamma}'(t))dt &= \int_0^1 \mathcal{F}(\gamma(1 - t), -\gamma'(1 - t))dt \\ &= \int_0^1 \mathcal{F}(\gamma(1 - t), \gamma'(1 - t))dt \\ &= \int_0^1 \mathcal{F}(\gamma(t), \gamma'(t))dt \end{aligned}$$

and by minimality of γ , $\tilde{\gamma}$ is indeed a minimal path from y to x and the values correspond : $d_{\mathcal{F}}(x, y) = d_{\mathcal{F}}(y, x)$. \square

Proposition 2.4. *If \mathcal{M} is compact, there is at least one curve γ achieving the minimum in (2.2).*

Proof. By Arzela-Ascoli theorem we have that $\{\gamma : [0, 1] \rightarrow \mathcal{M}, \gamma \in K - Lip\}$ is a compact set with respect to the uniform topology (because the sets $[0, 1]$ and \mathcal{M} are both compact and K -Lipschitz curves are equicontinuous). Taking a minimizing sequence, eventually reparameterizing so that every curve has constant speed equal to its length, we have that for any $\epsilon > 0$, $\forall n \geq n_0$ for some $n_0 \in \mathbb{N}$, $\mathcal{L}_{\mathcal{F}}(\gamma_n) \leq (1 + \epsilon)d_g(x, y)$.

Thus we have that

$$\forall s < t \in [0, 1], d_g(\gamma_n(s), \gamma_n(t)) \leq \int_s^t \|\gamma'_n(u)\|_g du \leq (1 + \epsilon)d_g(x, y)|t - s|,$$

which means that the sequence is compact for the uniform topology and thus converges up to a subsequence to a curve γ .

We conclude with the lower semicontinuity of the length functional that $\mathcal{L}_{\mathcal{F}}(\gamma) \leq \liminf_{n \rightarrow +\infty} \mathcal{L}_{\mathcal{F}}(\gamma_n) = d_g(x, y)$. \square

The map \mathcal{F} and distance $d_{\mathcal{F}}$ are usually called *Finsler metrics* and *Finsler distances* as opposed to the following Riemannian case.

Definition 2.5 (Riemannian Geodesic Distance). The Riemannian geodesic distance associated with the metric tensor field $g \in \mathcal{C}(\mathcal{M}, \mathcal{S}_d^{++})$, where \mathcal{S}_d^{++} denotes the set of symmetric definite positive matrices, is defined as follows :

$$d_g(x, y) = \inf_{\gamma \in \text{Lip}([0,1], \mathcal{M}), \gamma(0)=x, \gamma(1)=y} \int_0^1 \sqrt{g_{\gamma(t)}(\gamma'(t), \gamma'(t))} dt, \quad (2.3)$$

The following corollary is a direct consequence of proposition 2.3.

Corollary 2.6. *The functional d_g in Definition 2.5 is a distance equivalent to the euclidean norm when the metric tensor is continuous and bounded below (i.e. there exist $\alpha, \beta > 0$ such that $\alpha\|v\| \leq \sqrt{g_x(v, v)} \leq \beta\|v\|$).*

Proof. There are $\alpha, \beta > 0$ such that, $\forall v \in T_x \mathcal{M}, \alpha\|v\| \leq \sqrt{g_x(v, v)} \leq \beta\|v\|$, integrating over any curve $\gamma \in \Gamma_{x,y}, x, y \in \mathcal{M}$ and taking the infimum : $\alpha\|x - y\| \leq d_g(x, y) \leq \beta\|x - y\|$. \square

Note that the compactness of \mathcal{M} is a sufficient condition for the boundedness of g .

Proposition 2.7. *The restriction of a minimal path is also a minimal path.*

Proof. Let γ be a minimal path for the Finsler functional \mathcal{F} joining x to y . Then

$$\tilde{\gamma} : t \mapsto \gamma(t_0 + t(t_1 - t_0))$$

is an admissible path between $\gamma(t_0)$ and $\gamma(t_1)$. If it were to not be minimal, then we could find $\eta \in \text{Lip}([0, 1], \mathcal{M})$ with

$$\int_0^1 \mathcal{F}(\eta(t), \eta'(t)) dt < \int_0^1 \mathcal{F}(\tilde{\gamma}(t), \tilde{\gamma}'(t)) dt$$

and concatenation with γ would contradict minimality of γ . \square

The following result illustrates the link between the class of Riemannian metrics and the more general Finsler metrics (on \mathbb{R}^d) :

Proposition 2.8 (Braides et al. 2002). *Any Finsler distance defined on \mathbb{R}^d may be approached by a sequence of Riemannian distances, with smooth and isotropic metric tensor, in the uniform topology. Furthermore, the sequence of isotropic Riemannian length functionals approximate the Finsler length functional in the sense of Γ -convergence.*

Let's take a few easy examples to show what geodesic distances look like :

Example 1. The easiest example is the Euclidean manifold \mathbb{R}^d equipped with the metric tensor $g_x(u, v) = \langle u, v \rangle_{\mathbb{R}^d}$. Minimal paths are segments and the associated distance is simply the euclidean distance $d_g(x, y) = \|x - y\|_{\mathbb{R}^d}$.

Example 2. Another simple example that is important for this manuscript is the Riemannian isotropic metric tensor defined simply by giving a positive potential ϕ , thus defining $g_x(v, v) = \phi(x) \|v\|_{\mathbb{R}^d}^2$. This kind of isotropic model takes into account only the current position of the curve and not the orientation of the velocity vector. Geodesics derived from such a potential may be thought of as the trajectory of light in an isotropic milieu with Snell coefficient ϕ .

For what follows, the Eikonal equation and to have the ability to write *dual* problems related to the geodesic distances, it is useful to introduce the notion of dual to a metric :

Definition 2.9 (Dual Metric). The *dual* metric \mathcal{F}^* of a Finsler metric \mathcal{F} is defined by

$$\forall v \in T_x \mathcal{M}, \mathcal{F}^*(x, v) = \sup_{u \in T_x \mathcal{M}^*} \frac{\langle u, v \rangle}{\mathcal{F}(x, u)} \quad (2.4)$$

Example 3. In the Riemannian case with metric g , the dual metric is simply g^{-1} :

$$\mathcal{F}_g^* = \mathcal{F}_{g^{-1}}.$$

Let us show a small result ensuring the existence of a very handy parametrization under very few assumptions.

Proposition 2.10 (Constant Speed Parametrization). *For any curve $\gamma \in \mathcal{C}^1([0, 1], \mathcal{M})$ with nonzero velocity almost everywhere, there exist a reparametrization ϕ such that $\gamma \circ \phi$ has constant speed.*

Proof. Let $\psi : t \mapsto \frac{1}{\mathcal{L}_g(\gamma)} \int_0^t \|\gamma'(s)\|_g ds$, it is a \mathcal{C}^1 function mapping $[0, 1]$ to $[0, 1]$, strictly monotone, it is thus bijective with \mathcal{C}^1 inverse.

Defining $\phi = \psi^{-1}$ as the inverse of ψ , is is easy to verify that

$$\forall t \in [0, 1], (\gamma \circ \phi)'(t) = \psi^{-1'}(t) \gamma' \circ \psi^{-1}(t) = \mathcal{L}_g(\gamma) \frac{\gamma' \circ \psi^{-1}(t)}{\|\gamma' \circ \psi^{-1}(t)\|_g}. \quad (2.5)$$

It is now easy to see that $\gamma \circ \phi$ has constant speed equal to its length. \square

The following proposition ensures that we can recover the geodesic curves directly after the computation of the distance function.

Proposition 2.11. *Given $u : x \mapsto d_g(y, x)$, we can recover the geodesic curve joining y (for almost every $y \in \Omega$) to $x \in \Omega$ by solving the ODE :*

$$\begin{cases} \frac{d\gamma}{dt}(t) = -\frac{1}{d_g(x, y)} g^{-1} \nabla u(\gamma(t)), \\ \gamma(0) = x \end{cases} \quad (2.6)$$

Proof. Let γ be a length-parameterized geodesic from $\gamma(0) = x$ to $\gamma(1) = y$, then we have that $d_g(\gamma(t), y) = d_g(y, x) - d_g(y, x)t$. Taking the time derivative of the former, we have that $\frac{d\gamma}{dt}(t) \cdot \nabla_x u(\gamma(t)) = -d_g(y, x)$ i.e.

$$\frac{\sqrt{g}}{d_g(y, x)} \frac{d\gamma}{dt}(t) \cdot \sqrt{g^{-1}} \nabla_x u(\gamma(t)) = -1.$$

In the former euclidean scalar product, as both sides in the product are of euclidean norm 1 (on the right side because of the Eikonal Equation verified by the geodesic distance, see further section, and on the left side of the scalar product because of the parametrization of the curve we have that $\forall t \in [0, 1], \|\gamma'(t)\|_g = d_g(x, y)$), the case of equality in Cauchy-Schwarz allows us to conclude that

$$\frac{d\gamma}{dt} = -\frac{1}{d_g(x, y)} g^{-1} \nabla_x u(\gamma(t)),$$

that will be associated with the initial value $\gamma(0) = x$.

Having verified that a length-parameterized geodesic solves (2.6), we conclude by uniqueness. \square

Numerically the former ODE is quite easy to solve and stop when one has reached a small euclidean neighbourhood of target point y . What that means is that in general it is sufficient to compute the Riemannian distance via the methods shown in section 2.2

2.1.1 A very short introduction to Riemannian geometry

Let us define properly the notion of *Riemannian Manifold* and the vocabulary and tools that will be useful in the last chapter of this thesis.

Definition 2.12 (Riemannian Manifold). A *Riemannian Manifold* is a manifold \mathcal{M} equipped with a *Riemannian Metric* $g \in \mathcal{C}^2(\mathcal{M}, \mathcal{S}_d^{++})$.

Now let's define the Tangent Space to a manifold, which is where the derivatives and vector fields will live in the framework of Riemannian manifolds.

Definition 2.13 (Tangent space, Tangent Bundle). The tangent space $T_p\mathcal{M}$ to \mathcal{M} at point $p \in \mathcal{M}$ may be defined as the space of all velocities at that point of curves passing by p .

The *Tangent Bundle* $T\mathcal{M}$ is the space of couples (x, v) where $x \in \mathcal{M}$ and $v \in T_x\mathcal{M}$.

Now we define Vector fields formally.

Definition 2.14 (Vector fields). A vector field X is a map from the manifold \mathcal{M} to its tangent bundle i.e.

$$X : p \in \mathcal{M} \mapsto X(p) \in T_p\mathcal{M},$$

the set of all vector fields on \mathcal{M} will be noted $\mathfrak{X}(\mathcal{M})$.

Such vector fields define differentiations over the manifolds, but it is useful to define how those vector fields vary with respect to one another. This is the purpose of the following definition of Connections :

Definition 2.15 (Connection). A connection ∇ is a map from $\mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M})$ to $\mathfrak{X}(\mathcal{M})$ with the following properties :

letting $f, g \in \mathcal{C}^\infty(\mathcal{M})$ and $X, Y, Z \in \mathfrak{X}(\mathcal{M})$,

- $\nabla_X(Y + Z) = \nabla_X Y + \nabla_X Z$,
- $\nabla_{fX+gY}Z = f\nabla_X Z + g\nabla_Y Z$,
- $\nabla_X(fZ) = f\nabla_X Z + (Xf)Z$.

Usually we note Γ_{ij}^k its coordinates in a local frame (E_1, \dots, E_d) , that is to say

$$\nabla_{E_i} E_j = \sum_{k=0}^d \Gamma_{ij}^k E_k. \quad (2.7)$$

Among Connections that we may define over the manifold there are some that are more interesting than others. This is the case of the Levi-Civita connection which is one the most commonly used connection in Riemannian geometry :

Proposition 2.16 (Levi-Civita). *There is a unique connection ∇ that is both adapted to the metric g :*

$$\partial_X g(Z, Y) = g(\nabla_X Z, Y) + g(Z, \nabla_X Y), \quad (2.8)$$

and torsion-free :

$$\nabla_X Y - \nabla_Y X = [X, Y]. \quad (2.9)$$

Note that these two properties allow us to compute explicitly the coefficients of the connection (also called the Christoffel symbols), indeed the second equality gives us (taking $X = E_i$ and $Y = E_j$) the symmetry of the symbols $\Gamma_{ij}^k = \Gamma_{ji}^k$ and the first equation yields $\partial_k g_{ij} = \sum_m \Gamma_{kj}^m g_{mi} + \sum_m \Gamma_{ki}^m g_{jm}$

We may now define geodesics in the sense of Riemannian geometry

Definition 2.17 (Riemannian geodesic). A *Riemannian geodesic* is a curve parallel to itself or a straight curve, meaning analytically that it solves the following ODE :

$$\nabla_{\gamma'} \gamma' = 0 \iff \frac{D\gamma'_k}{dt}(t) + \sum_{0 \leq i, j \leq d} \Gamma_{ij}^k \gamma'_i(t) \gamma'_j(t) = 0, \quad (2.10)$$

Associated with initial conditions $(\gamma(0), \gamma'(0)) = (x, v)$, there is a unique maximal solution and we may define the *Riemannian Exponential map* as $\text{Exp}_x(v) \stackrel{\text{def.}}{=} \gamma(1)$, where $\gamma(1)$ is the unique solution evaluated at time 1 of the geodesic equation with initial values (x, v) when it is defined.

Let it be noted that this equation is the one that is obtained when applying Euler-Lagrange equations to the functional

$$\gamma \mapsto \int_0^1 g_{\gamma(t)}(\gamma'(t), \gamma'(t)) dt,$$

which is equivalent to looking at critical points of the length functional. This ensures that looking at Riemannian geodesics *for the Levi-Civita connection* means finding (locally) minimizing curves of length.

2.2 The Eikonal equation

The Eikonal equation is a non-linear first order Hamilton-Jacobi-Bellman PDE that describes how the geodesic distance functions behave. It is very important in this manuscript as it is essential to the computation of the distance functions and geodesic curves for the main methods used in imaging. Indeed it is at the basis of the numerical methods described in subsection 2.3 and subsection 2.4.

In the general case of a Finsler metric the eikonal equation may be written

$$\forall x \in \Omega \setminus S, \quad \mathcal{F}^*(x, \nabla u(x)) = 1, \quad (2.11)$$

with \mathcal{F}^* the dual metric of \mathcal{F} and S a chosen set where $u = 0$ is fixed.

In particular, for the case of a Riemannian metric g , the Eikonal equation becomes :

$$\forall x \in \Omega \setminus S, \quad \|\nabla u(x)\|_{g^{-1}(x)} = 1, \quad (2.12)$$

In the following we will expose in a bit more detail the theoretical foundation of the existence and uniqueness results for the solutions of the Eikonal equation. The first definitions presented are necessary to the definition of Viscosity solutions of PDEs.

Definition 2.18 (Tangent functions). • Let $u \in \mathcal{C}(\Omega, \mathbb{R})$, we say that $\phi \in \mathcal{C}^1(\Omega, \mathbb{R})$ is *tangent from above* at point $x \in \Omega$ if $\phi \geq u$ on Ω and $u(x) = \phi(x)$.

- We say that $\phi \in \mathcal{C}^1(\Omega, \mathbb{R})$ is *tangent from below* at point $x \in \Omega$ if $\phi \leq u$ on Ω and $u(x) = \phi(x)$.

Definition 2.19 (Degenerate Elliptic operators). Let L be an operator on $\mathcal{C}(\Omega, \mathbb{R})$, we say that it is *degenerate elliptic* if there exists a function $\mathcal{L} : (x, u, p, H) \in \Omega \times \mathbb{R} \times \mathbb{R}^d \times S_d \mapsto \mathcal{L}(x, u, p, H)$ such that

$$\forall u \in \mathcal{C}^2(\Omega, \mathbb{R}), Lu(x) = \mathcal{L}(x, u(x), \nabla u(x), \nabla^2 u(x)),$$

with \mathcal{L} non-decreasing with respect to its second variable and non-increasing with respect to its fourth (for the usual order on semi-definite positive matrices).

Definition 2.20 (Viscosity sub-/super-/ solutions). Let $u \in \mathcal{C}(\Omega, \mathbb{R})$ and L a degenerate elliptic operator.

- We say that u is a *viscosity sub-solution* of L if $\forall x \in \Omega, \forall \phi \in \mathcal{C}^2(\Omega, \mathbb{R})$ such that ϕ is tangent from above to u at x , $L\phi(x) \leq 0$.
- We say that u is a *viscosity super-solution* of L if $\forall x \in \Omega, \forall \phi \in \mathcal{C}^2(\Omega, \mathbb{R})$ such that ϕ is tangent from below to u at x , $L\phi(x) \geq 0$.
- u is a *viscosity solution* if it is both a viscosity super-solution and sub-solution.

Theorem 2.21. *There is a unique positive viscosity solution to the Riemannian Eikonal Equation (2.12) in (\mathbb{R}^d, g) , see Mirebeau 2023.*

Proof. (Viscosity Sub-solution) Let $v_0(x) = d_g(x_0, x)$ be the distance function from point $x_0 \in \bar{\Omega} \subset \mathbb{R}^d$. Let $x \in \Omega$, and let $\phi \in C^1(\Omega)$ be tangent superior to v_0 at x . Then for h small enough $x + h \in \Omega$ and

$$\phi(x) - \phi(x + h) \leq v_0(x) - v_0(x + h) \leq d_g(x, x + h) = \|h\|_g + o(\|h\|),$$

first applying the definition of a tangent from above function and then the triangular inequality.

Now sending h to zero yields $\|\nabla\phi(x)\|_{g^{-1}} \leq 1$.

(Viscosity Super-solution) Now let ϕ be tangent inferior to v_0 at $x \in \Omega$, let γ be a geodesic with unit parameteurization $\|\gamma'(t)\|_g = 1$ with $\gamma(0) = x$.

Using the following Taylor expansion for γ around 0:

$$\gamma(t) = x + \dot{x}t + o(t),$$

we now have

$$v_0(\gamma(t)) = v_0(x) - t,$$

and taking the Taylor expansion of the composition now :

$$\phi(\gamma(t)) = \phi(x) + t \langle \nabla\phi(x), \dot{x} \rangle + o(t),$$

and in the end :

$$t = v_0(x) - v_0(\gamma(t)) \leq \phi(x) - \phi(\gamma(t)) = -t \langle \nabla\phi(x), \dot{x} \rangle + o(t).$$

Noticing $|\langle \nabla\phi(x), \dot{x} \rangle| \leq \|\nabla\phi(x)\|_{g^{-1}} \|\dot{x}\|_g = \|\nabla\phi(x)\|_{g^{-1}}$, and combining with the former yields :

$$1 \leq \|\nabla\phi(x)\|_{g^{-1}}.$$

□

Remark. The existence and uniqueness result is true also for wider classes of Finsler Metric. One may refer to Bardi and Capuzzo-Dolcetta 1997.

2.2.1 The Fast Marching Algorithm

The Fast Marching method is the classic technique to compute geodesic distances on 2D or 3D domains in numerical applications. It is for instance ubiquitous in vessel tracking on medical images as we will see in Section 2.3.

Its first introduction is somewhat old, see J. A. Sethian 1996, but it has since seen many iterations Rickett and Fomel 1999; Popovici and J. A. Sethian 2002; Zhao 2004; Zhao 2006; Potter and Cameron 2021.

It is a single-pass method derived from Dijkstra's algorithm. The method is described in Algorithm 1. The idea is to start from the "Seed" set S and enumerate the nodes in the order of increasing distances until the "Target" set is reached or any other stopping criterion.

The key here is the estimation of the increment in the distance among neighbours on the grid : this is done using discretization schemes for the Eikonal Equation.

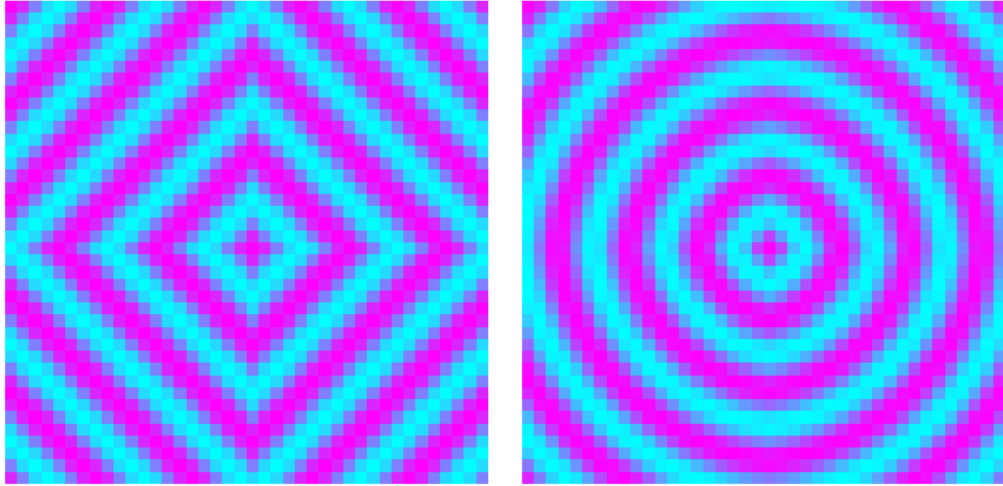


Figure 2.2: Level sets of the euclidean distance from the center of the grid computed with the Fast Marching algorithm, with scheme of order 1 (Left) and of order 2 (Right)

Example 4. In the case of isotropic metric tensor $g(x) = \phi(x)^2 I_d$, the Eikonal equation may be written $\|\nabla u\| = \phi$. The first idea for the local discretization is to apply finite differences directly : letting $a \in h\mathbb{Z}^d$ the "parent" of the point i considered in the grid, then the Eikonal equation becomes

$$\frac{(u_i - u_a)^2}{h^2} = \phi_i^2,$$

and thus

$$u_i = u_a + h\phi_i.$$

This case in particular corresponds immediatly to Dijkstra's algorithm on the grid with edge weights determined by ϕ . This approximation is too simple though, and introduces a huge bias in the computation of the distance on the grid (see Figure 2.2 left).

The upwind discretization scheme, introduced for the Eikonal equation in L. D. Cohen and Kimmel 1997, provides much more satisfying results for the Fast Marching algorithm :

$$\begin{cases} (u_p - u_{p \pm e_1})^2 + (u_p - u_{p \pm e_2})^2 = h^2 \phi_p^2 & \text{if } p \text{ has 2 parents,} \\ u_p = \min_i u_{p \pm e_i} + h\phi_p & \text{if } p \text{ has only 1 parent or } h^2 \phi_p^2 < (u_{p \pm e_1} - u_{p \pm e_2})^2. \end{cases}$$

This scheme is of order 2 and it allows us to compute distances with less bias due to the grid approximation of the space (see Figure 2.2 right).

Example 5. For general Riemannian metrics g in dimension d , we may use the eigenvalue decomposition method for positive definite operators. This way we have access to the eigenvalues and eigenvectors $(\rho_i(x), v_i(x))_{1 \leq i \leq q}$ of g_x at every point x from the grid. From this, the adaptation of the upwind scheme is direct :

$$\sum_{i=1}^d \frac{1}{h^2 \rho_i} \max(u_p - u_{p \pm v_i}, 0)^2 = 1.$$

Note that in general the $(v_i)_i$ in the former expression will not be supported on the grid, so one can resort to more sophisticated decompositions such as the Selling's decomposition or Voronoi's second decomposition, see Mirebeau 2019.

Figures 2.4 and 2.5 show resulting distance computations in the square 2D domain for the standard euclidean distance and the distance computed in a labyrinth via the Fast Marching method.

Algorithm 1 : Fast Marching Algorithm

Result : U , the geodesic distance from the set S

```

1 Initialization : each point in  $\Omega$  are labeled Far, the points in  $S$  are Accepted;
2 Initialization of  $U$  :  $U(x) = 0$  in  $S$ ,  $+\infty$  otherwise.
3 while Points are still labeled Trial or Far do
4   | Find a new point  $q$  minimizing  $U$  among the points labeled as Trial;
5   | Label  $q$  as Accepted;
6   | for  $p$  in the neighbourhood of  $q$  labeled as Trial or Far do
7     | Update  $U(p)$ ;
8     | if  $p$  is labeled as Far then
9       | | Label  $p$  as Trial ;
10    | end
11  | end
12 end

```

The subgradient marching algorithm

The subgradient marching algorithm is an interesting part of the litterature, as it is now a well established method, but not so widely used in practice. We introduce the method here as it will be very interesting for one of our applications shown in Section 3.4. It was first introduced in Benmansour, Carlier, et al. 2010.

Let d_ϕ be the geodesic distance on the plane associated with potential ϕ as in the former examples focusing on isotropic metrics. Once we know how to compute said geodesic distance, we may be interested in optimizing the choice of ϕ for certain tasks. To tackle such optimization or inverse problems, one needs to study the properties of the map associating a metric tensor to its corresponding geodesic distance function.

Proposition 2.22. $\forall x, y \in \Omega, \phi \mapsto d_\phi(x, y)$ is a concave map.

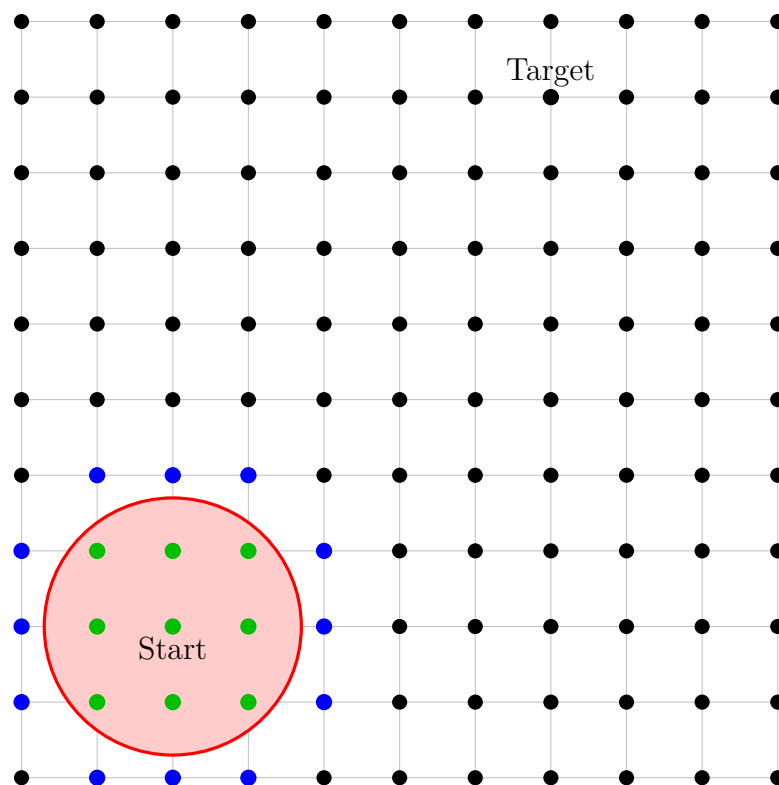


Figure 2.3: Representation of the Fast Marching Algorithm. Green points : nodes already Accepted. Blue points : nodes marked Trial. Black points : nodes that are not visited yet.

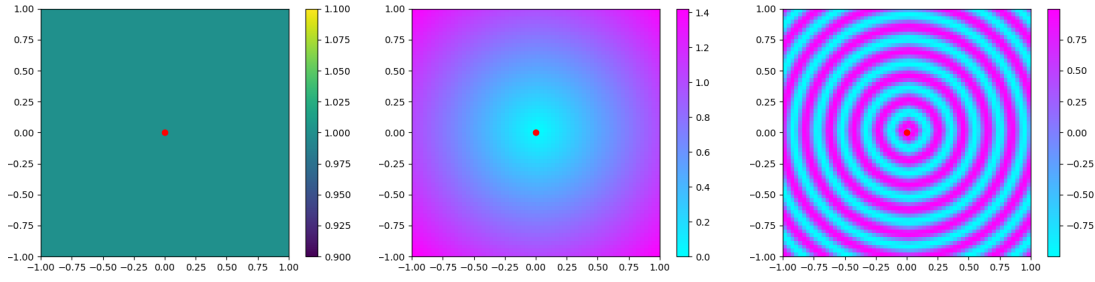


Figure 2.4: Fast Marching algorithm in Euclidean case. Left : Potential used for computation of (Isotropic) geodesic distance. Center : Distance field computed. Right : sinus of the Distance field to show level-sets.

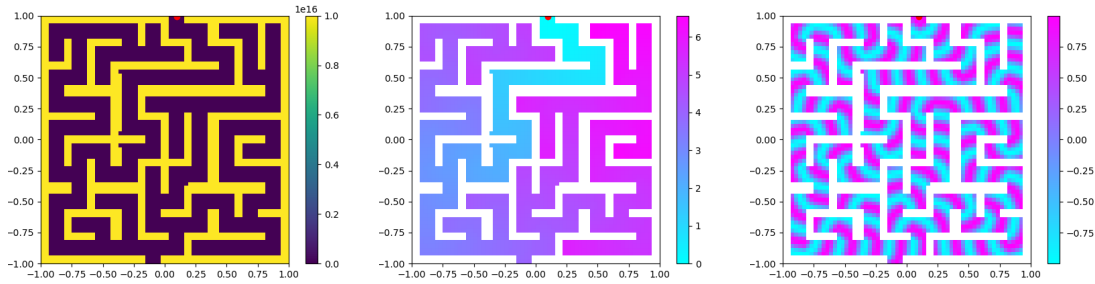


Figure 2.5: Fast Marching algorithm in Labyrinth. Left : Potential used for computation of (Isotropic) geodesic distance. Center : Distance field computed. Right : sinus of the Distance field to show level-sets.

Proof. Simply $d_\phi(x, y) = \inf_{\gamma \in \Gamma_{x,y}} \{\int_\gamma \phi\}$ is the infimum over a class of linear forms. \square

We can derive a "super-gradient" (that will be called "subgradient" again, by imitation of convex analysis) for such a concave function by perturbing ϕ and differentiating :

$$\left. \frac{dd_{\phi+\epsilon\xi}(x, y)}{d\epsilon} \right|_{\epsilon=0} = \int_{\gamma^*} \xi,$$

where γ^* is a minimizer for the ϕ -length functional.

It is then sufficient to extract a geodesic curve to have access to an element of the subdifferential, but it is sometimes hard to get in a robust discretized manner, which is why Benmansour, Carlier, et al. 2010 propose this alternative method.

Instead of discretizing the continuous expression of the subdifferential, they simply first consider the approximation of the distance given by the Fast Marching Algorithm described above, and then they differentiate through the iterations.

Differentiating with respect to ϕ in the two cases of update, we get

$$\begin{cases} (u_p - u_{p\pm e_1})(D_\phi u_p - D_\phi u_{p\pm e_1}) + (u_p - u_{p\pm e_2})(D_\phi u_p - D_\phi u_{p\pm e_2}) = h^2 \phi_p \\ \text{if } p \text{ has 2 parents,} \\ D_\phi u_p = D_\phi u_{p\pm e_i} + h \mathbb{1}_p \text{ if } p \text{ has only 1 parent or } h^2 \phi_p^2 < (u_{p\pm e_1} - u_{p\pm e_2})^2, \end{cases} \quad (2.13)$$

with $\mathbb{1}_p \in \mathbb{R}^{n^2}$ the vector filled with zero except at coordinate p , which gives the update:

$$\begin{cases} D_\phi u_p = \frac{(u_p - u_{p \pm e_1})D_\phi u_{p \pm e_1} + (u_p - u_{p \pm e_2})D_\phi u_{p \pm e_2} + h^2 \phi_p}{(u_p - u_{p \pm e_1}) + (u_p - u_{p \pm e_2})} & \text{if } p \text{ has 2 parents,} \\ D_\phi u_p = D_\phi u_{p \pm e_i} + h \mathbb{1}_p & \text{if } p \text{ has only 1 parent or if } h^2 \phi_p^2 < (u_{p \pm e_1} - u_{p \pm e_2})^2, \end{cases} \quad (2.14)$$

This update can then be used to compute the gradient of the geodesic distance with respect to the metric tensor during the Fast Marching iterations, with time complexity in $\mathcal{O}(n^2 \log(n))$ (as in (2.14), n additions along the $\mathcal{O}(n \log(n))$ iterations of Fast Marching).

2.2.2 Heat Method

The Heat Method is another method to approximate numerically the solution to the Eikonal equation, it leverages mathematical properties relating the heat equation to the Eikonal equation. It was recently popularized by the work of Crane et al. in Crane et al. 2013, which proposed a simple adaptation of the Heat Method. The method was then extended to anisotropic Riemannian metrics and Finsler metrics in Yang and L. D. Cohen 2016; Yang, Chai, et al. 2018.

The main idea of the method comes from the works of Varadhan 1967b and Varadhan 1967a where the author has studied respectively the asymptotic behaviour of the solutions of the heat equation when time nears 0 and the behaviour of the fundamental solution of the elliptic heat equation as transition probability as time-scaling goes to 0. Those two works have very close results, but they use two different approaches, one analytic and the second, with a more general result, probabilistic.

Indeed, let $p : (t, x, y) \mapsto p(t, x, y)$ be the solution to the heat equation, for some $x \in \Omega$,

$$\forall (t, y) \in I \times \Omega, \quad \frac{\partial p(t, x, y)}{\partial t} = \nabla_y \cdot (D(y) \nabla_y p(t, x, y)), \quad (2.15)$$

$$p(0, x, \cdot) = \delta_x, \quad (2.16)$$

where ∇_y is the gradient with respect to the variable $y \in \Omega$, then we have the asymptotic behaviour :

$$\lim_{t \rightarrow 0^+} -2t \log p(t, x, y) = d_{D^{-1}}(x, y)^2. \quad (2.17)$$

This formula is very useful as it links the geodesic distance to the solution of a very general PDE that is well known and whose solutions can be approximated accurately on a variety of domains. As pointed out in Crane et al. 2013, this result is to be understood as selecting trajectories of brownian motions that arrive at point y from point x in very short time i.e. with very little deviation from the minimal path. And more importantly than the formula itself, it is the fact that p and $d_{D^{-1}}$ follow the same gradients that is useful.

Indeed, directly using formula $d_{D^{-1}}(x, y) \simeq \sqrt{-2t \log p(t, x, y)}$ will often yield numerical instabilities, and the great value from Crane et al. 2013 comes from the

idea of using only the gradient of the solution to the heat method, and in a second time taking advantage of the Eikonal equation to recover the norm of the field.

This method is very practical because it boils down the problem to solving a numerical PDE and discretizing classic operators on a given domain. For instance the discretization of the Laplacian on the grid is a very well studied problem and we have many available discretizations for the grid, and different kinds of boundary conditions (see for instance Atkinson and W. Han 2009 for reference on numerical analysis). For surfaces and discretized manifolds in general it is also very easy to compute an approximation of the geodesic distance along the manifold as long as one knows how to discretize the Laplacian (see Figure 2.8 for an example of such computation on a mesh).

Such discretized operators are usually represented by sparse matrices which is very useful as we can take advantage of well-studied factorizations and make great economies in terms of memory and computational time.

Figures 2.6 and 2.7 show resulting distance computations in the square 2D domain for the standard euclidean distance and the distance computed in a labyrinth via the Heat method.

Figure 2.8 show the computation of the geodesic distance on the mesh of an elephant in 3D via the Heat method.

Algorithm 2 : Heat Method

Result : u^* , solution to (2.12)

- 1 **Initialization :** initialize δ_S with S the set from which we want to compute the distance, choose n the number of steps in the backward Euler scheme, t the time limit, Δ_D the Laplacian matrix associated to the metric tensor D
 - 2 Compute $\phi = (I - \frac{t}{n}\Delta_D)^{-n}\delta_S$
 - 3 Let $X = \frac{\nabla\phi}{\|\nabla\phi\|}$
 - 4 Solve $u^* \in \arg \min_u \|\nabla u - X\|_{L^2}^2$
 - 5 **return** u^*
-

Algorithm 2 sums up the different steps of the algorithm described in Crane et al. 2013 for the Heat Method.

Example 6. An easy example to look at to understand (2.17) is when solving on the euclidean space : we know that (2.15) can be solved using the convolution with a gaussian kernel, thus

$$p(t, x, y) = \frac{1}{\sqrt{4t\pi}} \exp\left(\frac{-\|x - y\|^2}{4t}\right),$$

and the Varadhan formula is not only true in the asymptotics but is exact at all times:

$$-4t \log p(t, x, y) - 2t \log(4t\pi) = \|x - y\|^2,$$

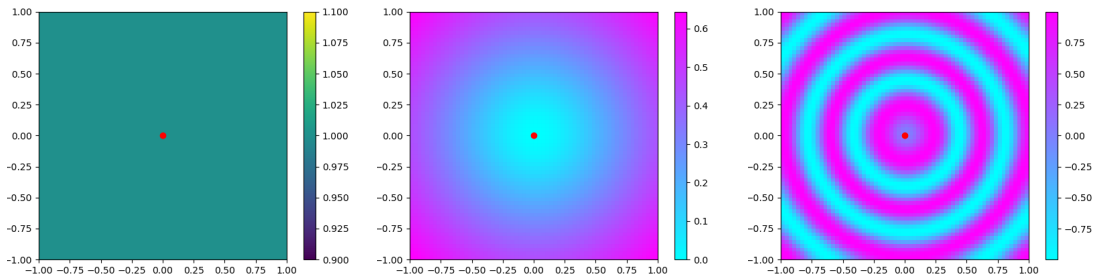


Figure 2.6: Heat Method algorithm in Euclidean case. Left : Potential used for computation of (Isotropic) geodesic distance. Center : Distance field computed. Right : sinus of the Distance field to show level-sets.

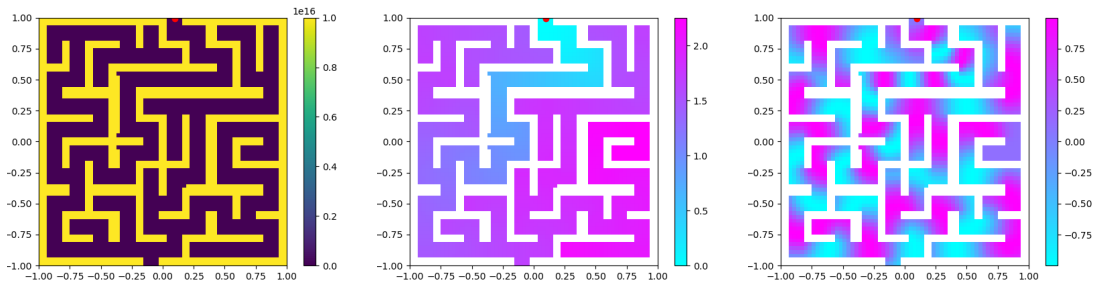


Figure 2.7: Heat Method algorithm in Labyrinth. Left : Potential used for computation of (Isotropic) geodesic distance. Center : Distance field computed. Right : sinus of the Distance field to show level-sets.

2.2.3 Other methods

Multiple other algorithms have been proposed to compute the geodesic distance on finite dimensional domains.

For instance in the recent work Ennaji et al. 2022, the authors propose to use dual formulations of the geodesic distance in order to compute it via a classic primal-dual algorithm.



Figure 2.8: Geodesic distance computed on a 3D mesh of an elephant.

Indeed, geodesic distance may be reformulated in a dual problem :

$$\sup_{u \in \mathcal{C}^0} \left\{ \int_{\Omega} u, \quad \forall x, y \in \Omega, \quad u(x) - u(y) \leq d_{\mathcal{F}}(x, y), \quad u = 0 \text{ on } S \right\}, \quad (2.18)$$

and equivalently

$$\sup_{u \in W^{1, \infty}} \left\{ \int_{\Omega} u, \quad \forall x \in \Omega, \quad \mathcal{F}^*(x, \nabla u(x)) \leq 1, \quad u = 0 \text{ on } S \right\}, \quad (2.19)$$

with S a set of points from which we compute the distance map.

These types of problems are named Beckmann problems.

As stated before, we have efficient algorithms to solve such problems. Reformulating as a minimization problem and putting constraints as indicatrix, we can rewrite :

$$\inf_{u \in W^{1, \infty}} \left\{ \int_{\Omega} -u + \chi_{B_{\mathcal{F}^*}}(\nabla u), \quad u = 0 \text{ on } S \right\}, \quad (2.20)$$

where $B_{\mathcal{F}^*} = \{v \in T_x \Omega, \quad \mathcal{F}^*(v) \leq 1\}$.

This is a minimization problem of the form

$$\inf_u \mathcal{E}(u) + \mathcal{G}(\nabla u), \quad (2.21)$$

which can be put into a primal-dual form

$$\inf_u \sup_{\phi} \mathcal{E}(u) + \langle \phi, \nabla u \rangle - \mathcal{G}^*(\phi), \quad (2.22)$$

The classic Primal Dual Algorithm can be easily rewritten in the context of our problem :

Algorithm 3 : Primal Dual Algorithm

Result : u^* , solution to (2.22)

- 1 Initialization : take $u_0 = \hat{u}_0$, $\phi_0 = \nabla u_0$ and choose $\eta, \tau > 0$
- 2 **for** $k \leq k_{max}$ **do**
- 3 $\phi_{k+1} = \mathbf{Prox}_{\eta\mathcal{G}^*}(\phi_k + \eta\nabla\hat{u}_k)$;
- 4 $u_{k+1} = \mathbf{Prox}_{\tau\mathcal{E}}(u_k - \tau\nabla^*\phi_{k+1})$;
- 5 $\hat{u}_{k+1} = 2u_{k+1} - u_k$
- 6 **end**

Using Moreau's identity we deduce the **Prox** operator related to \mathcal{G}^* :

$$\mathbf{Prox}_{\eta\mathcal{G}^*}(\psi) = \psi - \eta\mathbf{Proj}_{B_{\mathcal{F}^*}}(\psi/\eta)$$

and the computation for \mathcal{E} is easy :

$$\mathbf{Prox}_{\tau\mathcal{E}}(v) = (v + \tau)\mathbb{1}_{\Omega \setminus S} + u_S\mathbb{1}_S.$$

From what precedes, the reader may see that numerical computations are entirely dependant on our ability to compute the projection map associated to $B_{\mathcal{F}^*}$.

Classic results in the theory of numerical methods in convex analysis ensure that Algorithm 3 converges towards a solution to (2.22). See for instance Chambolle and Pock 2011.

Figures 2.9 and 2.10 show resulting distance computations in the square 2D domain for the standard euclidean distance and the distance computed in a labyrinth via the Heat method.

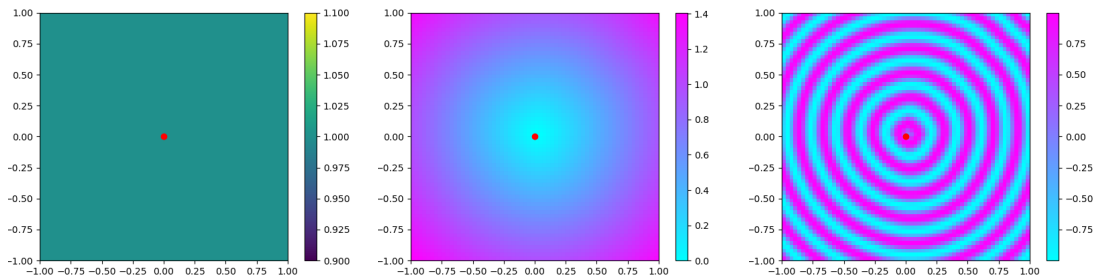


Figure 2.9: Primal Dual algorithm in Euclidean case. Left : Potential used for computation of (Isotropic) geodesic distance. Center : Distance field computed. Right : sinus of the Distance field to show level-sets.

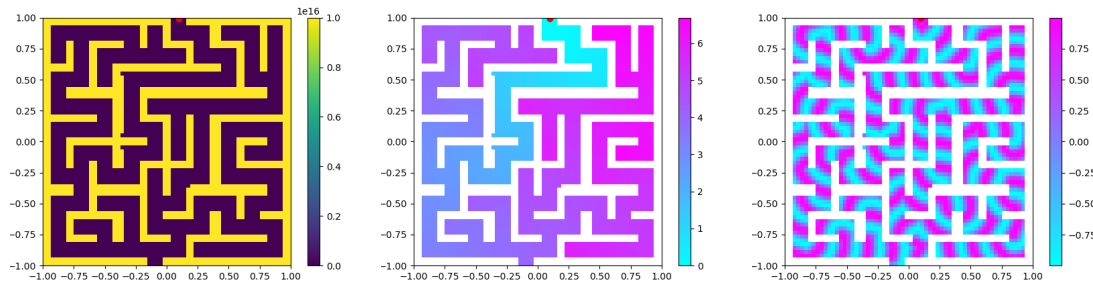


Figure 2.10: Primal Dual algorithm in Labyrinth. Left : Potential used for computation of (Isotropic) geodesic distance. Center : Distance field computed. Right : sinus of the Distance field to show level-sets.

2.3 Geodesic Methods in Image Processing

There is quite a long history now of geodesic methods being used to tackle segmentation, tracking of vascular structures and other image processing tasks. Here we may provide a short tour of existing methods involving geodesic methods, but the main resource around this subject is Peyré, Péchaud, et al. 2010.

Geodesic methods play a crucial role in image segmentation and object recognition by providing a robust framework for capturing the intrinsic geometry of visual data. Geodesic distances enable the delineation of object boundaries, facilitating more accurate and context-aware segmentation. This is particularly beneficial in scenarios with complex or irregular shapes where traditional methods may struggle.

Among the first works involving geodesic distance in applications, one may cite L. D. Cohen and Kimmel 1997, they use the Fast Marching Method in order to minimize an energy and find contours, boundaries or roads in natural images.

Li and Yezzi 2007; Benmansour and L. D. Cohen 2011; D. Chen, Mirebeau, and L. D. Cohen 2016; D. Chen, Mirebeau, Shu, et al. 2023 are among the first works to propose augmenting the dimension of the problem in order to add information, here by adding vessel width information. E. Bekkers et al. 2014; Duits, Meesters, et al. 2018 use orientation lifts in order to avoid shortcut problems in vessel tracking tasks.

L. Cohen 2001 allows one to segment regions by enclosing it inside the union of multiple well chosen geodesics.

Tasks in 3D have already been tackled for instance in Deschamps and L. D. Cohen 2001; Ardon and L. D. Cohen 2006, for vessel tracking in 3D.

Figure 2.11 shows an application of the process described in E. Cohen et al. 2018 with ULM data lifted in the space of position and orientation as done in Section 3.2, reimplementing this method in the peculiar context of ULM imaging was one of the first idea during our journey.

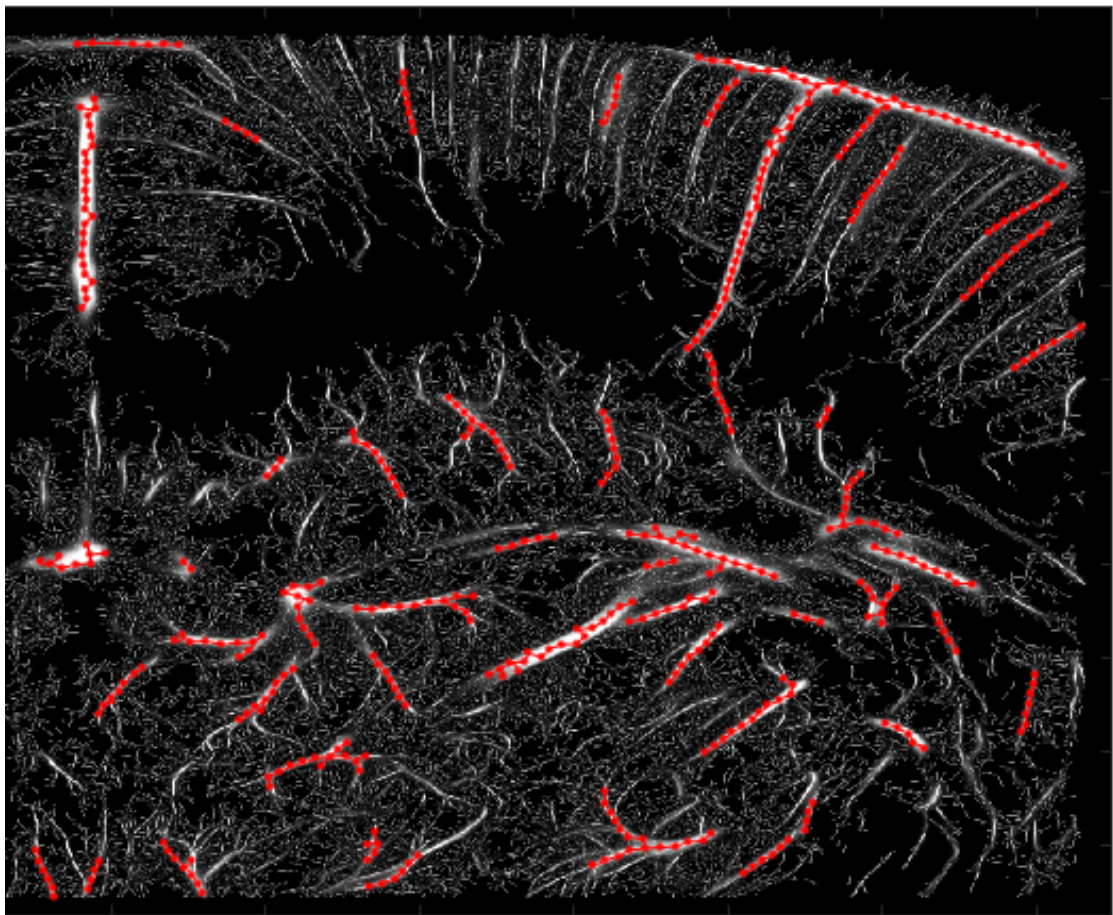


Figure 2.11: Segmentation method from E. Cohen et al. 2018 applied on ULM rat brain image with isotropic cost computed via Jerman filter.

Machine Learning for Computer Vision

Machine Learning and Deep Learning are nowadays the main paradigm to treat many problems that used to be mainly tackled from an Inverse Problem perspective. This is due to accessibility to a very large quantity of data and the will to produce it by practitioners of different fields in order to take advantage of the nice properties of models relying on data. It is also the availability of easy-to-use and relatively efficient autodifferentiation engines such as PyTorch or Tensorflow that allowed wider access to these Machine Learning techniques for a wide range of application fields.

3.1 Machine Learning, Deep Learning : First Principles

In the following we will introduce briefly the main principles of Machine Learning and related tasks.

3.1.1 Supervised Learning

Depending on the task at hand, the theoretical framework of Machine Learning (ML) problems may vary widely. For now, let us consider only tasks that are said to be *supervised*.

A ML task is said to be *supervised* when one has access to the theoretical answer to the problem that one tries to solve. Generally, the task will be set as follows : the practitioner has access to a number (finite or infinite) of observable variables $\{x_i\}_{i \in I} = X$ (which can be seen as realizations of a random variable) and the goal is to predict the associated labels $\{y_i\}_{i \in I} = Y$. As we try to be as general as possible, we do not specify the properties of the spaces of X and Y . The role of the ML practitioner is to determine a class of functions $\{f_\theta\}_{\theta \in \Theta}$ parameterized by θ living in a parameter space Θ , that will be appropriate to map the observable variables $\{x_i\}_{i \in I}$ to the labels $\{y_i\}_{i \in I}$.

Usually the supervised setting allows one to have access to a finite number of both variables and labels. The couple (X, Y) is called the *dataset* and it is almost always split into a *training set* and a *validation set*.

Once the model f_θ is defined, we want our model to have the right values for the examples known in our dataset; namely we want $\forall i \in I, f_\theta(x_i) \approx y_i$. In practice, it is not exactly what one wants, and we will rather only try to approach in some sense the labels on our dataset, as ML attempts to achieve models that have a more interesting property than simply interpolating the data : we want our model to be good at *generalizing* i.e. to be able to give good predictions for new data which was not available during training. Here we see the need to form a *validation dataset* : we want to avoid using all the data available for the computation of a good set of parameters θ^* , this way we can control if our trained model f_{θ^*} is able to generalize beyond data available at training time and thus avoid simply interpolating data. Often the dataset will be split one more time into a *test set* along the training and validation set, as the practitioner may want to modify the class of functions $\{f_\theta\}$ according to the performance evaluated on the validation set (we may speak of *meta-optimization*), in which case the optimal parameters are not independent of the validation set and fair evaluation requires that we have a third set.

The search for the optimal set of parameters is then performed by trying to fit the predicted values $f_\theta(x_i)$ to the labels y_i , which is done by minimizing a notion of discrepancy between the predictions and the labels.

The training process is then performed by minimizing some objective function

$$\min_{\theta \in \Theta} \sum_{(x,y) \in (X,Y)} L(f_\theta(x), y),$$

where (X, Y) is the training set of variable-labels pairs, and L is an appropriate function that compares the predictions to the labels (usually some notion of distance). One may sometimes add a regularization term $R(\theta)$ implementing some a priori on the parameters and the form of the solution.

One may interpret this minimization problem as the Maximum Likelihood estimation of the optimal parameter θ , with

$$p(\theta|(x, y)) \propto p(\theta)p((x, y)|\theta),$$

and

$$p(\theta) \propto \exp(-R(\theta)),$$

and

$$p((x, y)|\theta) \propto \exp(-L(f_\theta(x), y)).$$

Then Maximum Likelihood and our objective are equivalent

$$\arg \min_{\theta} \sum_{(x,y) \in (X,Y)} L(f_\theta(x), y) + R(\theta) = \arg \max_{\theta} \prod p(\theta|(x, y))$$

Classical loss functions include for instance :

- (p-norms) $L(z, y) = \|y - z\|_p^p$, $z, y \in \mathbb{R}^d$, $p \in [1, +\infty]$,
- (Binary Cross Entropy) $L(z, y) = (1 - y) \log(1 - z) + y \log(z)$,
- (Kullback-Liebler Divergence) $L(z, y) = y \log\left(\frac{y}{z}\right)$,

Once our framework is set up, one can notice for instance that some of the most classical statistical tasks may be put into this framework. For instance the choice $f_\theta(x) = \theta^T f(x)$ where $f : x \in \mathbb{R}^d \mapsto \mathbb{R}^{d'}$ and $L(z, y) = \|z - y\|_2^2$ yields the Linear Regression, additionally the regularisation term $R(\theta) = \|\theta\|_1$ yields the LASSO model.

One may note that the objective function is not convex in general, and as we will see next, the architectures f_θ defined are, more often than not, very far from making it convex. For models that are very far from the simple linear one, the general understanding of how everything works is "The optimization landscape is very complex, there are many local minima, but they are all *relatively good*".

3.1.2 Optimization : automatic differentiation and gradient descent

Automatic Differentiation (AD) is a computational technique widely used in numerical optimization, machine learning, and scientific computing for efficiently and accurately computing derivatives of functions. The main objective of AD is to evaluate the derivatives of a given function with respect to its input variables, which is essential for tasks like gradient-based optimization.

AD is grounded in the fundamental principles of calculus, specifically the chain rule. The chain rule states that the gradient of a composition of functions is the product of the Jacobian and/or gradients of those functions. In the context of AD, this rule is systematically applied to decompose complex functions into a sequence of elementary operations.

Written for the composition of two differentiable functions $f : \mathbb{R}^m \rightarrow \mathbb{R}$ and $g : \mathbb{R}^d \rightarrow \mathbb{R}^m$, the chain rule reads :

$$\forall x \in \mathbb{R}^d, \frac{\partial(f \circ g)}{\partial x_i}(x) = \sum_j \frac{\partial f}{\partial y_j}(g(x)) \frac{\partial g_j}{\partial x_i}(x). \quad (3.1)$$

Automatic Differentiation is a procedure to compute recursively the derivative of a function with respect to some input, the idea being that the function in question should be expressed as the composition of many simple functions. Usually, the different steps to compute a function of the inputs with respect to which the function is to be derived are stored in a computational graph. The graph is then read backwards to compose the gradients.

The step of computing the objective function is called the *forward pass* and the second step of computing the gradient is called the *backward pass*.

The concrete implementation in PyTorch requires each function called to have both a **forward** and **backward** method implemented, at the backpropagation step each **backward** method is called recursively and the result is multiplied with the former **backward** calls from further down in the graph.

3.1.3 Neural Network architectures

The successes of Machine Learning in recent years comes both from the rise in access to a big amount of data, the accessibility and efficiency of automatic differentiation libraries and the development of a wide variety of architectures of Neural Networks.

Indeed, in the setup presented above, a type of approximating class in particular has received a lot of attention lately, that is the Neural Network. Originally thought of as a replica of the way signals propagate through the brain and neural plasticity, Neural Networks try to mimic the travel of an input signal into a network of neurons. That is done by applying successive linear transformations, whose coefficients are parameters to be optimized (they model the way neurons interact between each other), and non-linear transformations also called *activation functions* (thought as the way neurons communicate or not to the next neurons depending on whether the input intensity attains a certain threshold level or not). To stick to the biological analogy, activation functions used to be chosen as some kind of approximation of a Heaviside function such as sigmoids, arctan or tanh, but experience has shown that using simpler ("least non-linear") activation functions such as the ReLU = $\max(0, \cdot)$ and its variations gives better results (in part because of their non-zero gradient away from 0).

Let $n \in \mathbb{N}$, $(h_k)_k$ be a set of activation functions and $\forall 1 \leq i \leq n, W_i, b_i$ be a collection of weights and biases, in general Neural Networks with n layers taking input in \mathbb{R}^d can be written as :

$$\forall x \in \mathbb{R}^d, f_\theta(x) = h_n(W_n h_{n-1}(W_{n-1} h(\dots h_1(W_1 x + b_1) \dots) + b_{n-1}) + b_n). \quad (3.2)$$

Such architectures are usually called *Multilayer Perceptrons*, and most neural network architectures may be written like this, or as a combination of such perceptrons or with additional structural conditions over the weights. This architecture is interesting because we have theoretical guarantees that, for different sets of conditions over their width and depth, and hypothesis over the activation functions, they can approximate any continuous function. This class of theorems are called *Universal Approximation Theorems*. Figure 3.1 illustrates one *neuron* i.e. one of the units in the former composition.

Equivariant networks

Some of the Neural Networks architectures that receive the most attention in the literature have particular properties of invariance, or rather equivariance, to specific transformation groups.

Convolutional Neural Networks (CNN) were proposed and have since revealed to be very useful in various signal processing tasks and other ML tasks. The main idea of this architecture is to have each of the linear transformations in (3.2) be

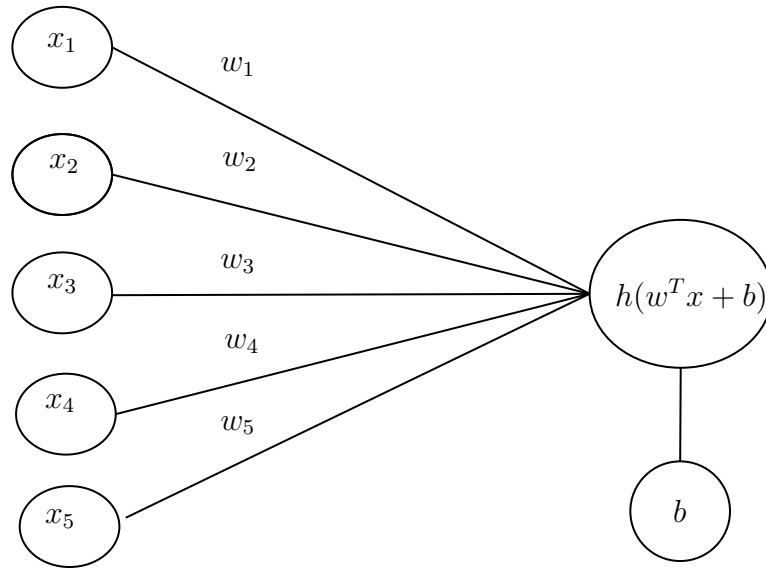


Figure 3.1: Illustration of a theoretical neuron, the x_i are the inputs, w_i are weights, b is a bias and h is the activation function.

convolutions with a kernel function (which is usually to be learnt from data). We may cite the foundational works LeCun et al. 1989; W. Zhang et al. 1990 who were the first to apply such an architecture to computer vision problems.

Let us write such a transform for a signal defined on an open set of the real line (representing e.g. an audio signal) :

Let $f : \mathbb{R} \rightarrow \mathbb{R} \in L^2(\mathbb{R})$, be our signal the linear transform $W : L^2(\mathbb{R}) \rightarrow L^2(\mathbb{R})$ defined by the kernel $k \in L^2(\mathbb{R})$,

$$\forall x \in \mathbb{R}, \quad Wf(x) = \int_{\mathbb{R}} k(x-y)f(y)dy, \quad (3.3)$$

the group related to these linear transforms is that of translations of the real line. This group acts on \mathbb{R} and on $L^2(\mathbb{R})$ through $\tau_u x = x - u$ and $\tau_u f(x) = f(\tau_u x) = f(x - u)$, and it is easy to see that $\forall u \in \mathbb{R}, W\tau_u f = \tau_u Wf$. Choosing such linear transforms ensures the network to be *equivariant* (i.e. transformation of the input results in the same transformation of the outputs, as long as each of the transformations in the network is equivariant). This is interesting because it allows to reduce the dimension of the features to be learned : for instance the detection of a subject in a part of an image is independent from the location of the subject in the image. Figure 3.2 illustrates the equivariance property.

One may note that the discrete counterpart to (3.3) fits into the framework of (3.2) :

$$(Wf)_i = \sum_j k_{i-j} f_j, \quad (3.4)$$

as this discretized operator may be written as a specific type of matrix, namely Toeplitz matrices. Indeed a Toeplitz matrix A is defined by the equation :

$$\forall i, j, \quad A_{ij} = k_{i-j}.$$

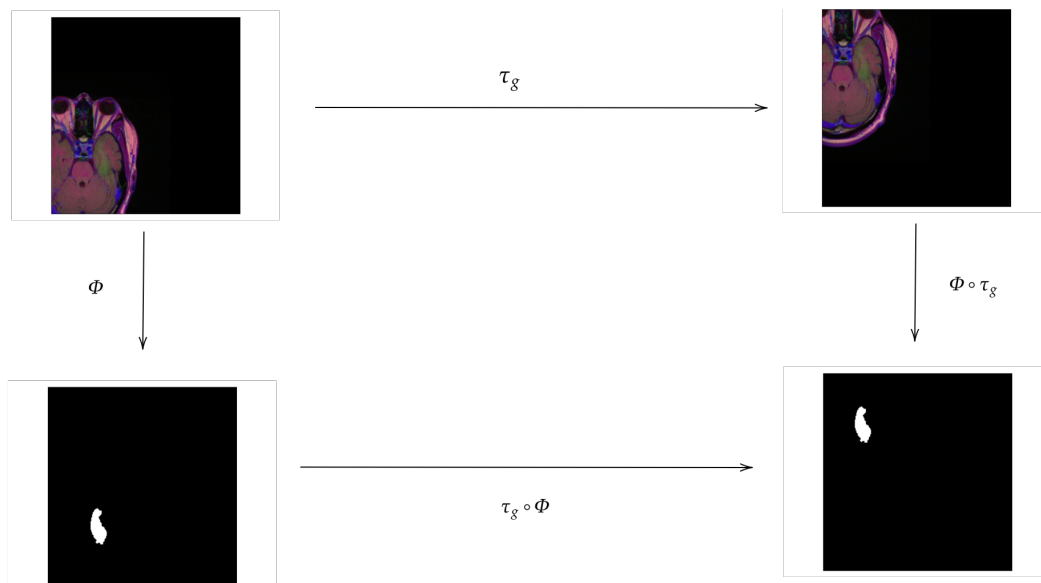


Figure 3.2: Illustration of the Equivariance property for translation applied to a brain tumour segmentation task (see dataset Pedano et al. 2016). The map Φ is equivariant with respect to the action of the group G if the diagram commutes.

One might want to note that the ReLU function is also equivariant for translations :

$$\forall x \in \mathbb{R}, \quad \max(0, f(x - u)) = \max(0, f)(x - u).$$

In recent years there have been many works focusing on extending this setup to new group structures and have equivariant architectures available. We refer to the foundational works T. S. Cohen and Welling 2016; T. Cohen and Welling 2016; T. S. Cohen, Geiger, et al. 2018 on equivariant CNN architectures and the book Bronstein et al. 2021 for an overview of the subject. We may also cite Smets et al. 2023 for a different approach considering each linear transform as applying an equivariant PDE on an input.

The Encoder-Decoder architecture

A very common way to conceptualize Neural Networks is to think of them as learning representations of data with low intrinsic dimension but living in high dimensional spaces. The Auto Encoder expands on this notion by having two successive branches, the first "Encoder" branch takes data as input in a high dimension space and sends it to a lower dimensional space (one that we can think near the intrinsic dimension of the data), and a second "Decoder" branch that sends this representation back to the original high dimensional space.

The theoretical space described by the decoder is called the *latent space*. It is extremely useful when there is no natural way to describe the data, for instance this concept has seen many successful applications in unsupervised learning applied

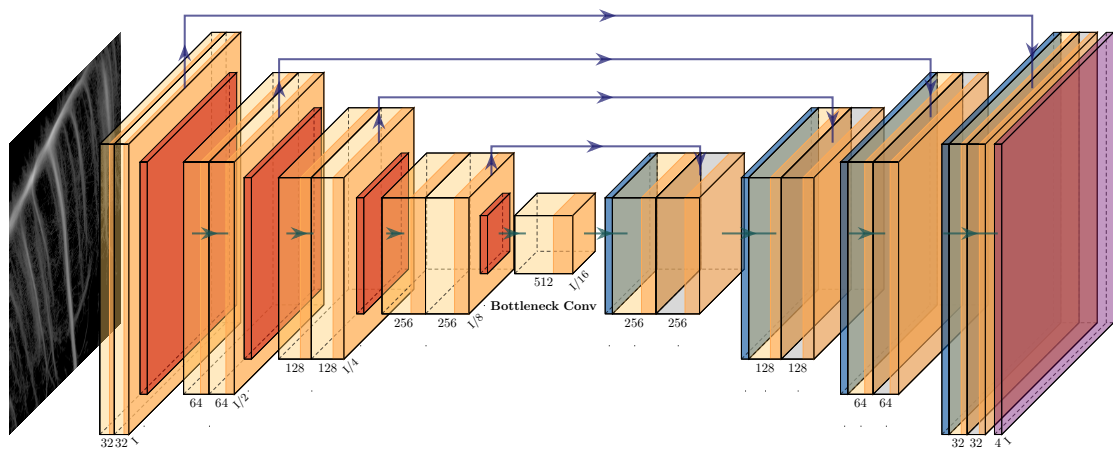


Figure 3.3: UNet CNN architecture.

to Natural Language Processing. One of the main problematic in the processing of language is to be able to give a useful and efficient representation of chains of characters in a way that conveys meaning. Once a proper representation of words is possible, one can simply see speech as walking in the abstract space that accurately represents language.

The UNet Architecture

The UNet architecture is a CNN model that is widely used in Computer Vision tasks and in Biomedical Imaging treatments. It reproduces the classical Encoder-Decoder architecture with small convolutional layers, maxpooling operations to reduce the size of the images along the network at the encoding stage, and up-convolution at the encoding stage. There are also skip-connections between the encoding layers and the decoding layers to pass large scale information that get lost during the encoding phase, reducing dimensionality of the features. Figure 3.3 represents the UNet architecture.

3.2 Tracking vascular trees on medical images

This is a joint work with Laurent D. Cohen. Accepted at the IMPROVE 2024 conference, it has been published online as part of the Proceedings of the 4th International Conference on Image Processing and Vision Engineering (IMPROVE 2024).

3.2.1 Introduction

As stated in the Introduction of this manuscript, ULM is a novel imaging technique that allows users to bypass the compromise between precision and depth of penetration in ultrasound imaging.

It allows one to make highly resolved images of the vascular network deeper in the skin tissues with the help of micro bubbles used as contrast agents. We refer to Couture, Hingot, et al. 2018 for an overview of the super resolution method.

In the present work, we introduce a new workflow for complete end-to-end detection of vascular structures on ULM images, using deep learning to detect landmarks (see Figure 3.4) allowing tracking of vessels as edges in a tree graph with landmarks as vertices. Our approach differs from classical Perceptual Grouping for blood vessel tracking as performed in E. J. Bekkers, D. Chen, et al. 2018; Benmansour and L. D. Cohen 2009a. Indeed we are trying to take advantage of long geodesics tracking blood vessels across an image, that should behave well given the amount of literature on the subject. While Perceptual Grouping usually focuses on computing short geodesics between close points spread across the vessel network, we aim to compute few long geodesics between key landmarks of the vasculature. We try to take advantage of the information specifically given by ULM imaging, but it must be noted that it is possible to adapt on other types of images, for instance eye fundus images obtained via direct photography. To do so, one may need to evaluate local orientation information as we will see in section 3. From a 2D image it can be done using Orientation Scores as in Duits, Felsberg, et al. 2007 or similar transforms such as the ones presented in J. Zhang et al. 2016 to lift a 2D image set in the plane to the 3-dimensional space of positions and orientations.

Detection, segmentation or tracking tasks on medical images are widely studied problems.

The contributions of our work include :

- working with ULM data, defining a Riemannian metric in order to track vessels in ULM images,
- dealing with scarcity of data : 2 different high resolution images to make both the training and validation set,
- carrying out detection of vascular landmarks in such context,
- fitting a tree model with geodesics as edges to take into account geometric and topological aspects into the tracking, thus investigating the efficiency of using the tree-like nature of vasculature to perform the tracking,
- comparing results on synthetic data (hand-made black and white images to

fit the framework used for ULM data, i.e. few big images) and eye fundus images (more images, but smaller).

Vessel segmentation is usually performed by computing scores of vesselness on the image, see the seminal work Frangi et al. 1998 and more recent work Jerman et al. 2016. The main idea in these works being that high vesselness corresponds to regions in the image where one orientation is dominant. Vesselness is then defined as a function of the eigenvalues of the Hessian (in dimension 2, one eigenvalue being significantly higher than the other indicates a tubular region). Modern methods of transposing the image in a higher dimensional setting of Position-Orientation space were used in J. Zhang et al. 2016.

Other methods of vascular segmentation include machine and deep learning methods that have become accessible thanks to the availability of annotated data. We can cite for instance Oliveira et al. 2018 that uses a fully-convolutional U-net for segmentation task on eye fundus images. Even more recent ideas include adding Attention mechanisms inside neural network architectures as it is one of the main paradigm in Machine Learning these days. One may cite the Vision Transformer Dosovitskiy et al. 2021 and the Swin-UNet architecture as much more recent architectures used to tackle segmentation problems that may seem similar to the processing presented here. In fact the reader may note that here we are indeed realizing a tracking of the blood vessels, i.e. we are trying to find curves that represent accurately blood vessels present in the image, but it has to be noted that this is quite far from the most common approach in modern vessel detection methods. Indeed the problem usually considered in the literature is not to find vessels as curves but simply to indicate pixels in the image where the vessels are supposed to be. In Section 3.2.5 we attempt to give a good assessment of the performance of our method and to compare its performance against segmentation scores in the same fashion as it is commonly done in the literature. This is of course at the disadvantage of the geodesic methods as it is already a non trivial task to project and widen curves in order to get a segmentation mask from a parameterized curve.

A few works have already approached the problem of localizing vascular landmarks in eye fundus images Abbasi-Sureshjani et al. 2015; Pratt et al. 2017; Wang et al. 2023; Calvo et al. 2011; Tetteh et al. 2020 but they usually first focus on providing a segmentation mask of the blood vessels in the image before carrying post-processing on the segmentation to infer the positions of the landmarks (*end-points*, *crossings* or *bifurcations* of blood vessels). Hervella et al. 2019 tries to tackle the problem of finding landmarks directly from input data, and we will be building on this method here.

The second part of our workflow makes use of geodesic curves to track vascular structures in the input images. Tracking vascular structures using geodesics has been done multiple times for instance in Deschamps and L. D. Cohen 2000; Benmansour and L. D. Cohen 2011, those methods have had multiple extensions, for instance taking advantage of roto-translation group E. Bekkers et al. 2014 or adding vessel width information Li and Yezzi 2007.

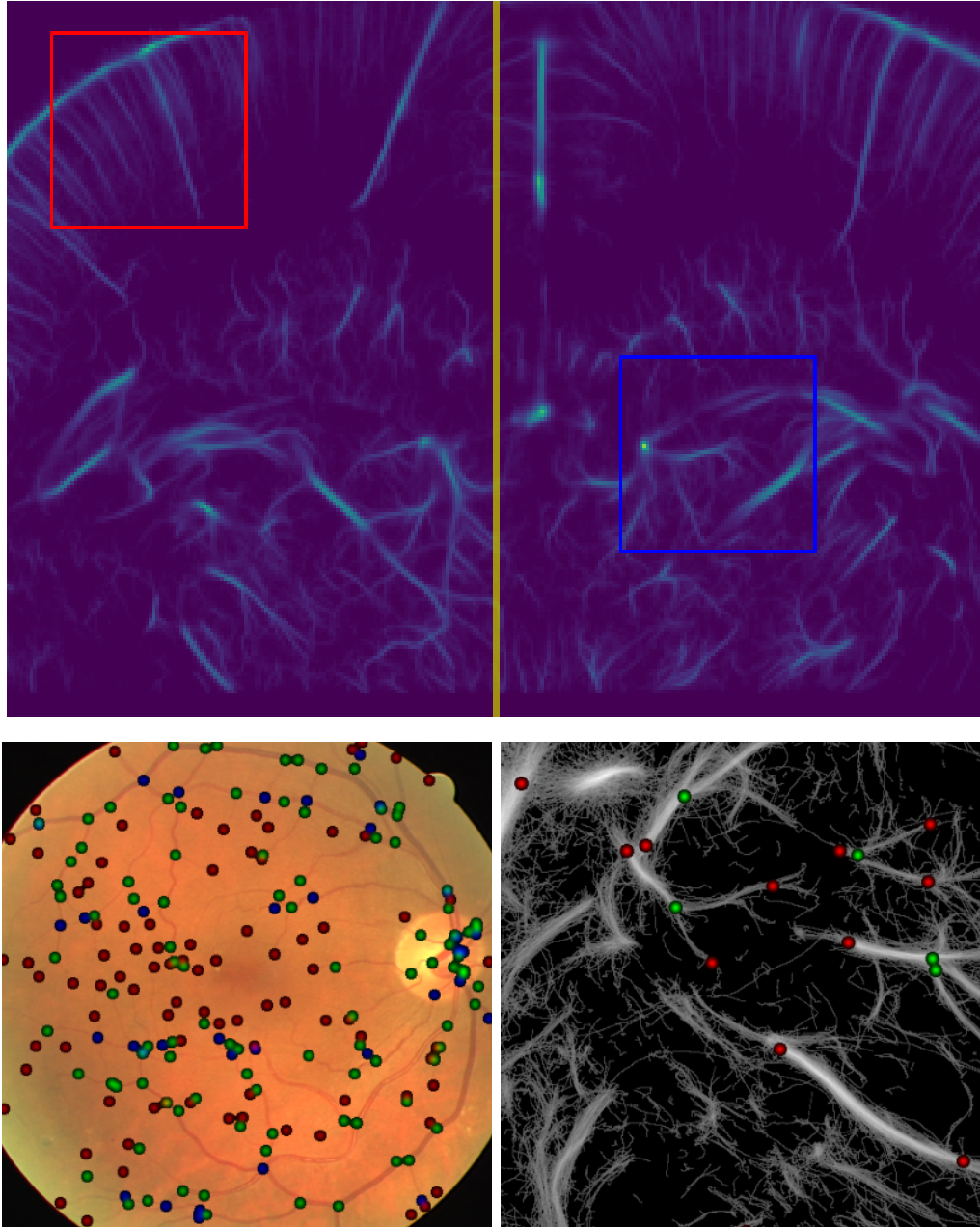


Figure 3.4: Top : Patches are made from a high resolution image by cropping patches taken uniformly from each brain half. Bottom Left : eye fundus image overlaid with heatmap of vascular landmarks. Bottom Right : ULM image overlaid with heatmap of vascular landmarks.

3.2.2 Detecting vascular landmarks

The approach to detect the vascular landmarks is very much like the one used in Hervella et al. 2019. The novelty of our approach is the scarcity of ULM data we use for learning, the integration of *endpoints* in the detection task, and the tracking described further down.

Indeed, we want to generate heatmaps with multiple channels indicating probable locations of vascular landmarks.

We train a single U-net architecture to learn the localization and classification of interesting points in a 2D ULM image of brain vessels. The network outputs 4 channels : 3 for different types of landmarks (*endpoints*, *bifurcations*, *crossings*) and a last one to relax classification, by predicting the maximum across all three other channels, we allow the network to miss the classification of the landmarks but still detect a point nonetheless. The heatmap predicted by the CNN is then filtered to get the position of local maxima (after thresholding at level r of output of network to reduce noise). See Figure 3.5 for an illustration of the thresholding and local maxima detection in the heatmap output by the CNN.

ULM Data

Our data is composed of few highly resolved images of rat brains obtained via ULM imaging. The scarcity of data is a usual problem in medical images processing. We were provided with two such images of rat brains (two similar plans were imaged) by Chavignon et al. 2020, we then proceeded to uniformly cut those big (3210×2675) images into smaller square patches. This way, we aggregate around 42 ULM images, making a training dataset of 21 and another set for validation of 21 images. We make sure that there are patches from both original images in both sets. We also make sure that there is no overlapping between the training and validation datasets by using different brain halves to make those patches (see Figure 3.4 top).

As there is no available dataset annotation for segmentation task of ULM data nor for the landmark localization and classification task, annotation for the latter was produced by one of the authors. The dataset annotation was made by selecting the point landmarks by hand with the appropriate tool Skalski 2019. One great difficulty of our approach is that we are highly dependent on the accuracy of the initial annotation of the data which can be hard given that there are multiple visible vessel sizes on ULM images.

To prove the efficiency of our method, we also apply it to two other datasets : one that consists in two synthetic images of tubular structures arranged into a network, for training and validation, they are very big and are used to imitate the case of ULM images (high resolution, few images, thus cut into smaller patches); the other one is simply a dataset using both images and groundtruth from the retina image DRIVE and IOSTAR datasets, much like in the previously cited work Hervella et al. 2019.

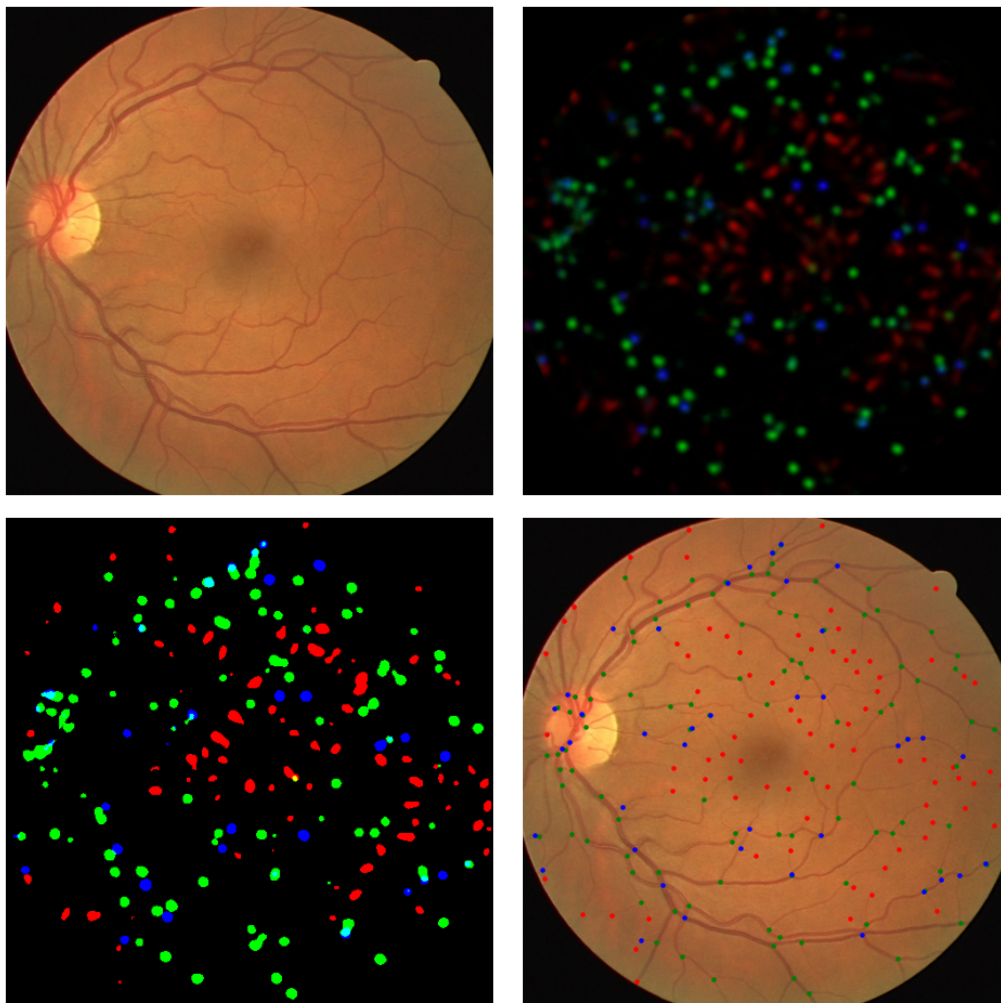


Figure 3.5: Illustration of the localisation process on one of the image from the training set. Top left : input image. Top right : output of the network. Bottom left : output of the network thresholded at the value 0.2. Bottom right : points detected as local maxima scattered over the input image. For the output and the scattered points, red indicates the *endpoint* class, green *bifurcation* and blue *crossing*.

Training

The training loss is defined as $\mathcal{L}(\theta, x, y) = \|f_\theta(x) - y\|_2^2$ the mean squared error (MSE), where y is the position of the labeled features in the input images in our dataset convolved with a gaussian kernel $y = \sum_{y \in D} k_\sigma * \delta_y$, f_θ is our CNN architecture with parameters θ , applied on the x input image. k_σ is a gaussian kernel with chosen standard σ .

Going through our data we minimize the loss evaluated on the training set over the space of parameters $\theta \in \Theta$. We may recall that the U-net architecture (see Figure 3.3 is an encoder-decoder architecture composed of multiple convolution layers (3×3 filters and leaky ReLU activation) and with skip-connections. Ronneberger et al. 2015 is the fundamental work introducing this architecture.

To make up for the small size of our dataset, we perform data augmentation via

horizontal symmetries, translations, rotations, all randomly applied with predefined parameters. It allows us to artificially expand our training dataset, leveraging equivariance of our task by the action of those transformations.

Evaluation Metrics

We evaluate the performance of our CNN on the validation set by evaluating Precision, Recall and F1-score.

We say that some point that was found is a True Positive (TP) if it is located at a distance to any labeled landmark less than some threshold λ_{tol} (that we usually take around 7 pixels width), and that it is a False Positive (FP) otherwise or if another point has already been detected in the vicinity of the same landmark. The same way a point in the labeled data set that was not found is said to be a False Negative (FN).

Then, we get that $Pr = \frac{TP}{TP+FP}$, $R = \frac{TP}{TP+FN}$ and $F1 = \frac{2Pr \cdot R}{Pr+R}$ is the harmonic mean between Precision and Recall.

3.2.3 Finding appropriate geodesics

Geodesics have been used for tracking vessels in vascular images for a long time now. The works Deschamps and L. D. Cohen 2000; Benmansour and L. D. Cohen 2011 laid good basis for such work, and the book Peyré, Péchaud, et al. 2010 that is a good introduction to the use of geodesics for image analysis. These works leverage our knowledge of geodesic curves and numerical algorithms allowing us to compute them to track tubular structures on medical images.

Geodesics for vessel tracking

Vessel tracking can be performed by finding geodesics on an image of the vessels. To do so, we simply need to define a metric that is well adapted.

For now, we will restrict ourselves to Riemannian metrics defined on the homogeneous space of positions and orientations $\mathbb{M}_d = \mathbb{R}^d \times \mathbb{P}^{d-1}$ with $\mathbb{P}^{d-1} \simeq \mathbb{S}^{d-1}/\{-1, 1\}$, \mathbb{P}^{d-1} allows us to assimilate features that have the same direction but not the same sign. we will use the relaxed Reeds-Shepp metric that is well-studied, Riemannian and penalizes curves that are not planar. Indeed the curves we want to recover should be planar i.e. such that (in the case $d = 2$) the local orientation coordinate θ corresponds to the orientation of the curve : let $\forall t \in [0, 1], \eta(t) = (x(t), y(t), \theta(t)) \in \mathbb{R} \times \mathbb{R} \times (\mathbb{R}/2\pi\mathbb{Z})$ then η is planar if and only if $\theta(t) = \arg(x'(t) + iy'(t))$. We can see that such planar curves are simply the curves that canonically are $2D$ curves lifted in the position-orientation space.

The original Reeds-Shepp model was built to model the movements of a car in the plane which means the movements considered are constrained by the orientation of the car and that it is only allowed for the car to move in the direction of the wheels represented by the orientation coordinate θ . The first model Reeds and Shepp 1990 completely forbids any movement orthogonal to the orientation of the car. It can only be considered in our framework as a sub-Riemannian model, and the original publication by Reeds and Shepp breaks down the optimal paths for such a car into

sequences of words ("go left", "go right" and "go straight"). The optimal trajectories described in this model have "cusps" i.e. times at which the trajectory's velocity has a discontinuity and it is with the ambition to build an approximation of the Reeds-Shepp car without one of these cusps that the relaxed Riemannian Reeds-Shepp car model was proposed in Duits, Meesters, et al. 2018.

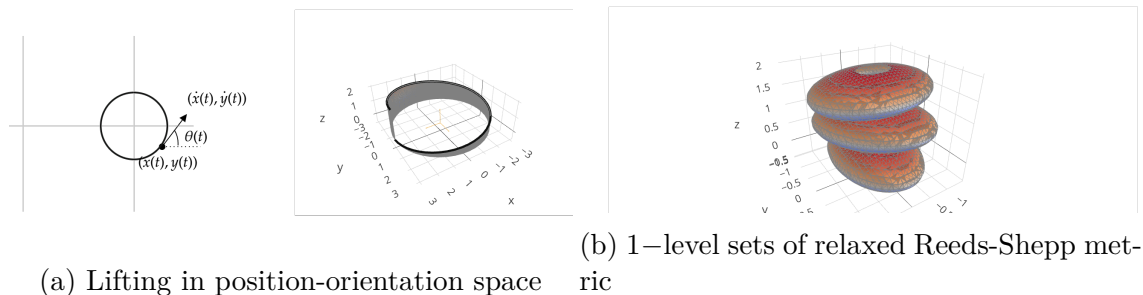


Figure 3.6: Illustrations of the Lifting in the position and orientation space and of the relaxed Reeds-Shepp metric (1-level sets of relaxed Reeds-Shepp metric in the tangent spaces at $\theta = 0, \pi/4, \pi/2$ for $C \equiv 1$, $\epsilon = 1/\sqrt{2}$ and $\xi = \sqrt{5}$).

The relaxed Reeds-Shepp metric is the one associated with the metric tensor defined by :

$$\mathcal{P}_\varepsilon((x, \theta), (\dot{x}, \dot{\theta}))^2 = C((x, \theta))^2 (|\dot{x} \cdot e_\theta|^2 + \frac{1}{\varepsilon^2} |\dot{x} \wedge e_\theta|^2 + \xi^2 |\dot{\theta}|^2), \quad (3.5)$$

with $\xi, \varepsilon \in \mathbb{R}$, e_θ the unit vector with orientation θ . C is a cost function, in the following it will be defined as

$$C = \frac{1}{1 + \lambda W^2}$$

with $\lambda = 10^3$ and W a $[0, 1]$ -valued score built from the image.

Figure 3.6 illustrates both the lifting of points in the orientation-lifted domain and the behaviour of the anisotropic part of the metric defined.

This vesselness score W is important because it allows us to associate a θ coordinate to all the detected landmarks by finding the orientation θ maximizing the score at its position. As detailed further down this section, the vesselness score for ULM images is built by making an histogram of the position and orientation and orientations in a discretized 3d space, see Figure 3.7 for a sample of such vesselness score.

We also look to enrich our orientation-dependent score by adding information from the detection of vascular landmarks by imposing $\forall \theta, W(x, \theta) = 1$ if a *bifurcation* has been detected at position x , so that the landmark point is accessible from any orientation. Figure 3.8 shows this on a real sample from ULM images.

Similarly, if a landmark has been found and classified as a *crossing*, we add a new point to our set of detected points located at this position but with second maximum intensity in the vesselness score.

Figure 3.9 shows an example of the final vesselness map used in (3.5).

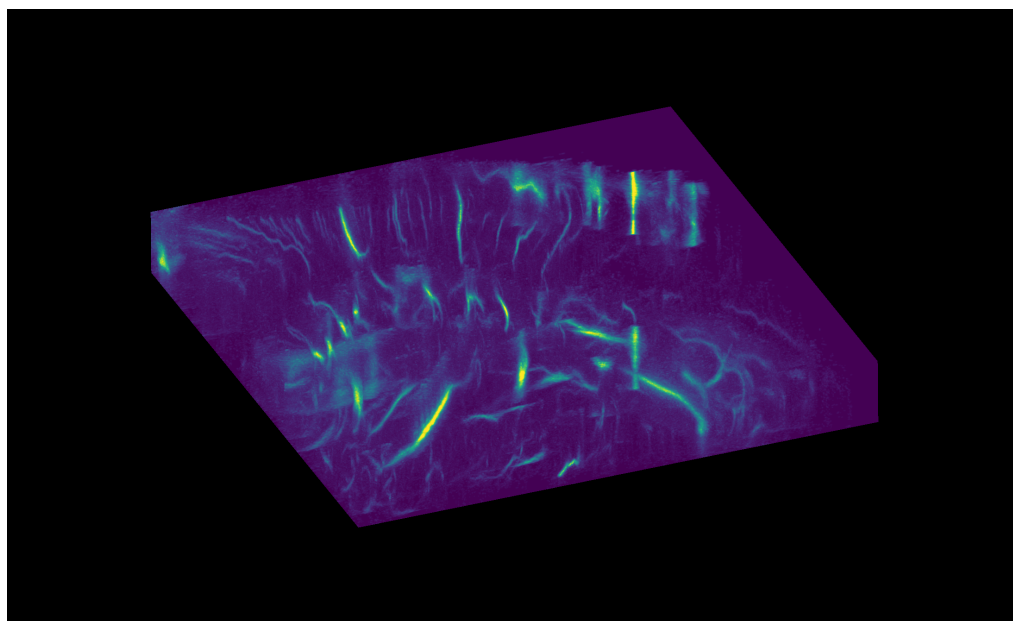


Figure 3.7: ULM image lifted in the space of positions and orientations, used as vesselness map in the distance model.

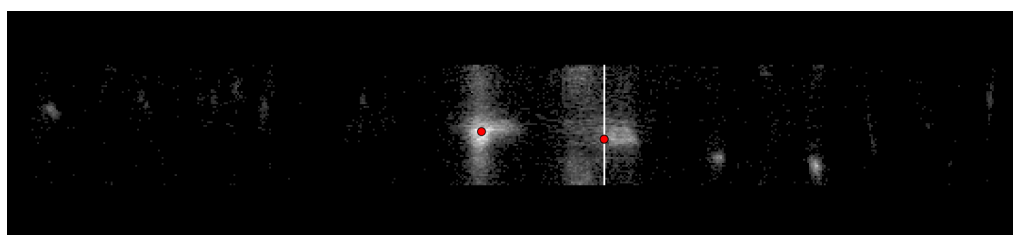


Figure 3.8: $X - \theta$ Slice of the final vesselness map used for the computation of the cost for ULM data in the distance model around a detected landmark classified as "bifurcation". The gray image in background is the vesselness map, red points are the detected points. As one can see, the red point on the right classified as "bifurcation" has allowed us to modify the vesselness map by saturating the map at the position of the point and for all orientations around this point.

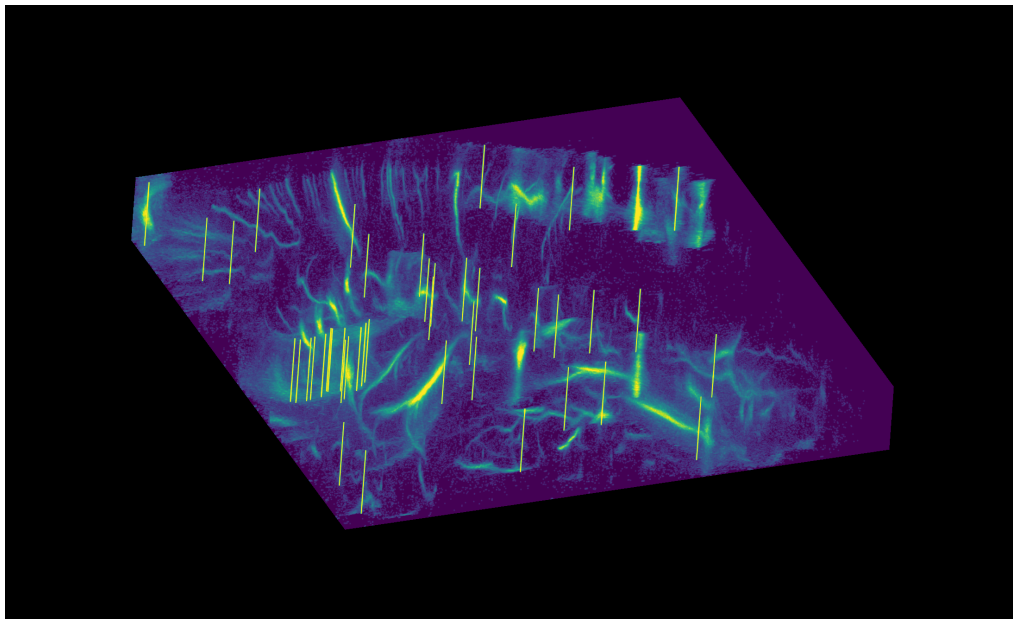


Figure 3.9: Final vesselness map on ULM image (shown in viridis colormap for easier visualization)

Geodesics defined through this relaxed Reeds-Shepp model are well-studied and have already been used in previous works to accurately track blood vessels (for instance in Duits, Meesters, et al. 2018). The main asset of this model is that it helps avoid shortcuts in the case where two different vessels cross in a 2D image. Also, as noted before, it is essential in order to have the geometry accurately allowing the retrieval of the connected landmarks as trees.

The geodesic distance can be computed efficiently and fast using the Fast Marching Algorithm, we refer to Duits, Meesters, et al. 2018 and the attached library for efficient computational tools used in the present work.

Clustering landmarks using the geodesic graph

Once we have defined a proper geodesic distance and we are able to effectively compute it, we can build a matrix $D = (d(x_i, x_j))_{1 \leq i, j \leq n_l}$ of pairwise distance, where the x_i are the n_l detected landmarks.

This step is the computational bottleneck as it requires to compute $n_l(n_l - 1)/2$ coefficients, meaning solving n_l times the Fast Marching algorithm to iteratively fill the lines of the matrix, computing the map $d(x_i, \cdot)$ at every iteration i . Thus the complexity is around $\mathcal{O}(n_l N \log(N))$ with N the number of points. Although it is a long computation to make, it has to be noted that it is very easily parallelizable which allows us to mitigate the impact on the time of inference.

The computed pairwise distance matrix thus defines a complete weighted graph that we call the geodesic graph. On a single image there may be many different groups of vessels that appear, with the computation of the pairwise distance we have already computed the geodesic curves between each pair of points. We then need to keep only the groups of points that are relevant for representation of the vessels. To cut the complete graph into smaller connected components, we will simply perform

hierarchical clustering on the graph. Indeed, if the metric is well chosen to make landmarks linked by the vascular network near for the distance d , and landmarks not connected by the vascular network far, we simply group aggregate points that are near and separate them from the others under some condition of threshold distance $s_{cluster}$.

Hierarchical Clustering : We will be performing Hierarchical clustering over our landmarks in order to group them into cluster. Here we go into a bit more detail on the Hierarchical Clustering method. Given a set of N elements $X = \{x_1, \dots, x_N\}$ and a distance d between those elements, we want to build classes C_i , subsets of X . We will build them recursively, and to do so, we need to define a dissimilarity metric between two given classes : common ways to define such metric are $diss_{\min}(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$, $diss_{\max}(C_1, C_2) = \max_{x \in C_1, y \in C_2} d(x, y)$ and $diss_{\text{mean}}(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{x \in C_1, y \in C_2} d(x, y)$.

We can notice that we have $diss_{\min} \leq diss_{\text{mean}} \leq diss_{\max}$, larger dissimilarity metric meaning smaller classes for a given distance threshold between classes in the end.

Let be N classes in the first step : $\forall 1 \leq i \leq N, C_i^0 = \{x_i\}$, using one of the dissimilarity seen before, we have $diss(C_i^0, C_j^0) = d(x_i, x_j)$.

We proceed to build recursively, from a given set of $0 \leq k < N$ classes $C_1^{N-k}, \dots, C_k^{N-k}$, a new set of $k - 1$ classes by simply taking $(i^*, j^*) = \arg \min_{(i,j)} diss(C_i^{N-k}, C_j^{N-k})$, we define $C_1^{N-k+1} = C_{i^*}^{N-k} \cup C_{j^*}^{N-k}$ and then add the other classes renamed as $C_i^{N-k+1}, 2 \leq i \leq k - 1$.

Initializing as seen, each of our step decreases the size of the problem by one, after N iterations we've constructed all $C_i^l, 0 \leq l \leq N, 1 \leq i \leq N - l$.

To choose which level of classes we want to keep, we can simply fix the number i.e. if we want k classes our clustering is given by the set $\{C_i^{N-k}, 1 \leq i \leq k\}$.

If we are agnostic relatively to the number of classes we need in the end, we simply use a threhsold $s_{cluster}$ and take $k^* = \max_{1 \leq k \leq N, \min(diss(C_i^{N-k}, C_j^{N-k})) \geq s_{cluster}} k$ i.e. the smallest number of classes such that classes are separated of at least $s_{cluster}$ for the chosen dissimilarity metric (if $\{1 \leq k \leq N, \min(diss(C_i^{N-k}, C_j^{N-k})) \geq s_{cluster}\}$ is empty, we take $k^* = 1$).

This algorithm simply corresponds to cutting the dendrogram tree at a given height (as illustrated in Figure 3.10).

We may cite the work Müllner 2011 as a reliable source for theory and algorithms for hierarchical clustering.

Linking landmarks through the geodesic graph

Our landmarks are now separated into multiple groups. Each of these group supposedly represents one connected component of the visible vascular network in the 2D image.

Those smaller groups represent smaller complete graphs, but we still need to select which of the computed geodesics represent the vessels.

Now the objective is to only keep the curves that accurately represent blood vessels in the image. This can be done by removing some of the edges in each smaller graph.

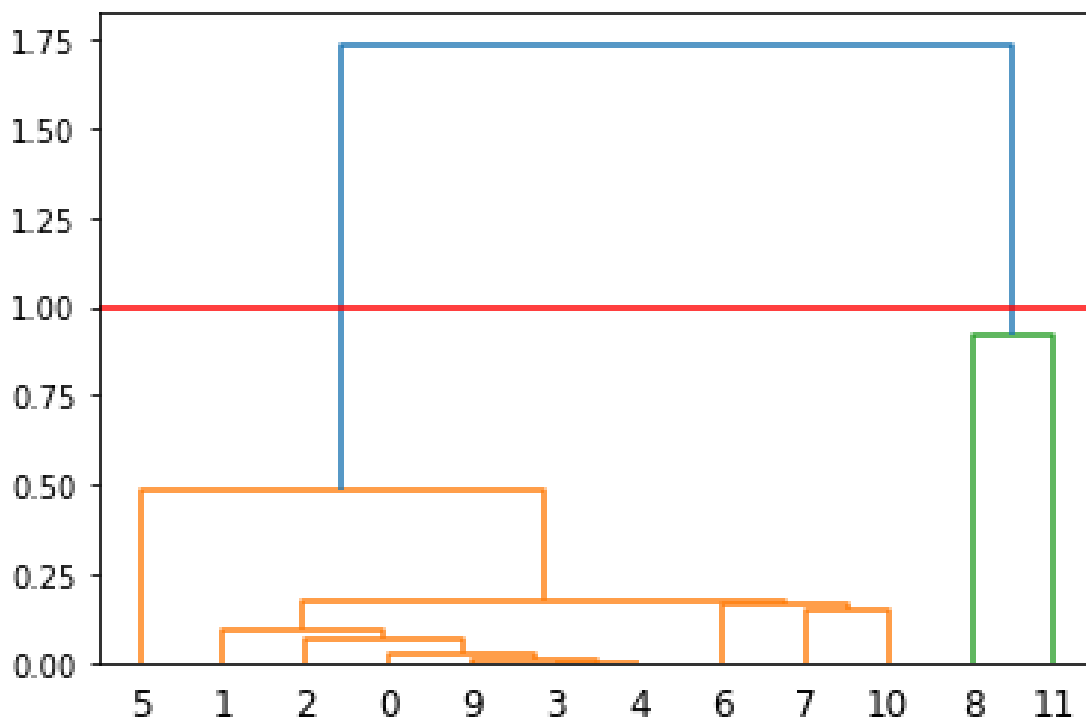


Figure 3.10: Example of the way we cut to define the clusters as groups below the threshold $s_{cluster} = 1.0$. Each index at the bottom represents one point and the dendrogram tree shows which pairs of points are clustered together. The red line shows the distance threshold at which we cut the tree, separating it into two connected components (yellow and green). The blue part of the dendrogram shows edges in the initial complete graph that are deleted after the clustering.

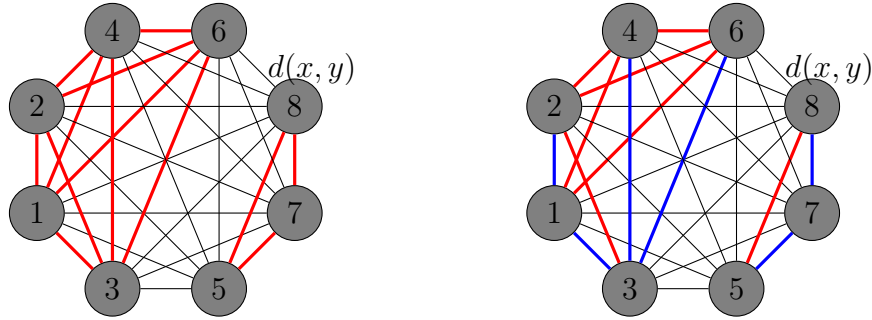


Figure 3.11: Illustration of the 2 steps on a 8–complete graph : in the first step we separate clusters of points (in red), an then we find the minimum spanning tree in each connected component (in blue).

We need a few properties for the target graph, inferred from the idea we have of the representation of blood vessels :

- As it should represent vessels, it needs to be "1–dimensional" i.e. it is represented by a planar graph.
- It does not have cycles.
- It is small for the geodesic distance (if previously well chosen metric).

A good heuristic to have these properties is to look for the minimum spanning tree in each smaller complete cluster graphs. The minimum spanning tree in a connected weighted graph is defined as a subset of the edges of the initial graph that defines a connected and acyclic subgraph, connecting every vertices and minimizing for the sum of the weights.

A few algorithms are available to perform this computation efficiently. For Kruskal’s algorithm, time complexity is of the order of the sorting of the edges’ weights $\mathcal{O}(e \log(e))$ with e the number of edges in the graph. With a complete graph we have $e = (V - 1)!$ with V the number of vertices, thus e can get very large. In general we don’t have a lot of vertices in each small graph, so it does not pose a problem to use Kruskal’s algorithm. For the sake of completeness, we cite Prim’s algorithm for which the time complexity is $\mathcal{O}(V^2)$. See Cormen et al. 2009 for further reading on the Minimum Spanning Tree problem and algorithms.

3.2.4 Results and Discussion

In this subsection we present the results of our method on our different datasets.

Synthetic data

To test our framework we can execute our algorithm on synthetic hand drawn data. The main difference is that if we use 2D images to simulate our network, there is no straightforward way to define orientation-lifted images as we do with ULM microbubble trajectories data. To generate such orientation-lifted images, we leverage our knowledge of Orientation score techniques.

We define a simple transformation by convolution with the help of anisotropic gaussian kernels : the anisotropy in a direction θ of the kernel will allow us to select

only the parts where the local features are aligned with this direction. The kernel is given by :

$$\forall x \in \mathbb{R}^2, \quad k_{\theta}^{\text{lift}}(x) \stackrel{\text{def.}}{=} e^{-\langle e_{\theta}^{\perp}, x \rangle^2 / \sigma^2} e^{-\|x\|^2 / \epsilon^2}$$

with e_{θ}^{\perp} the unitary vector with orientation perpendicular to θ , and σ, ϵ positive constants defining the level of anisotropy and the spread of the kernel (bigger support of the kernel means we get lower frequency features and blurring of discontinuities, whereas a small support retains higher frequency information).

This kernel defines the lifting operator Φ :

$$\forall u \in L^2(\Omega), \quad \forall (x, \theta) \in \Omega \times [0, \pi[, \quad u^{\text{lifted}}(x, \theta) = (\Phi u)(x, \theta) = (k_{\theta}^{\text{lift}} * u)(x),$$

As it can be seen in Figure 3.12, the image that we've used are very simple and try to imitate the behaviour of a vascular tree in smaller patches included in a bigger image (typically eye fundus image). This dataset will allow us to test our workflow and understand its behaviour in a very simple test case.

To train the CNN to detect landmarks, we use two synthetic images : one for the training set and the other for the validation and apply the same approach as in the case of ULM data.

Results : At the landmark-detection task level, we were able to attain an F1 score of about 77% on the validation dataset with hyper-parameters selected by hand and *recall* scores of 80% and 75% respectively) as evaluated on the whole validation image. During the training process, validation is made on a set of images taken as random crops from the original big image. The validation scores obtained on those smaller (256×256) patches tend to give similar scores on average along the dataset, although the scores oscillate a lot along epochs. Figure 3.13 shows the resulting tracking of synthetic vessels and the corresponding geodesic graph.

Eye fundus image

To reinforce our methodology, we apply our workflow to a classical dataset of eye fundus images, as a middle ground between synthetic and ULM data.

The data came from the IOSTAR and DRIVE dataset Abbasi-Sureshjani et al. 2015, it was split into a training set (30 images), a validation set (10 images) and a test set (21 images).

For the training, we perform random data augmentation with translations and rotations, as to avoid overfitting and take advantage of equivariance properties of the task at hand, and also random crops of fixed size.

We built the vesselness score W for the computation of minimal paths by first applying a classical Frangi filter Frangi et al. 1998 on the input images and then lifting the filtered images via Orientation Score as described in the previous subsection. See Figure 3.14 for an example of the application of such a vesselness filter and its lifting in the position-orientation space.

Results : We were able to achieve satisfying results of about 60% in F1 score on both validation dataset and test dataset (after hyper-parameter searching on the validation data). These results on the landmarks detection task is not as good as the ones presented in Hervella et al. 2019 but they predict only two classes of landmarks

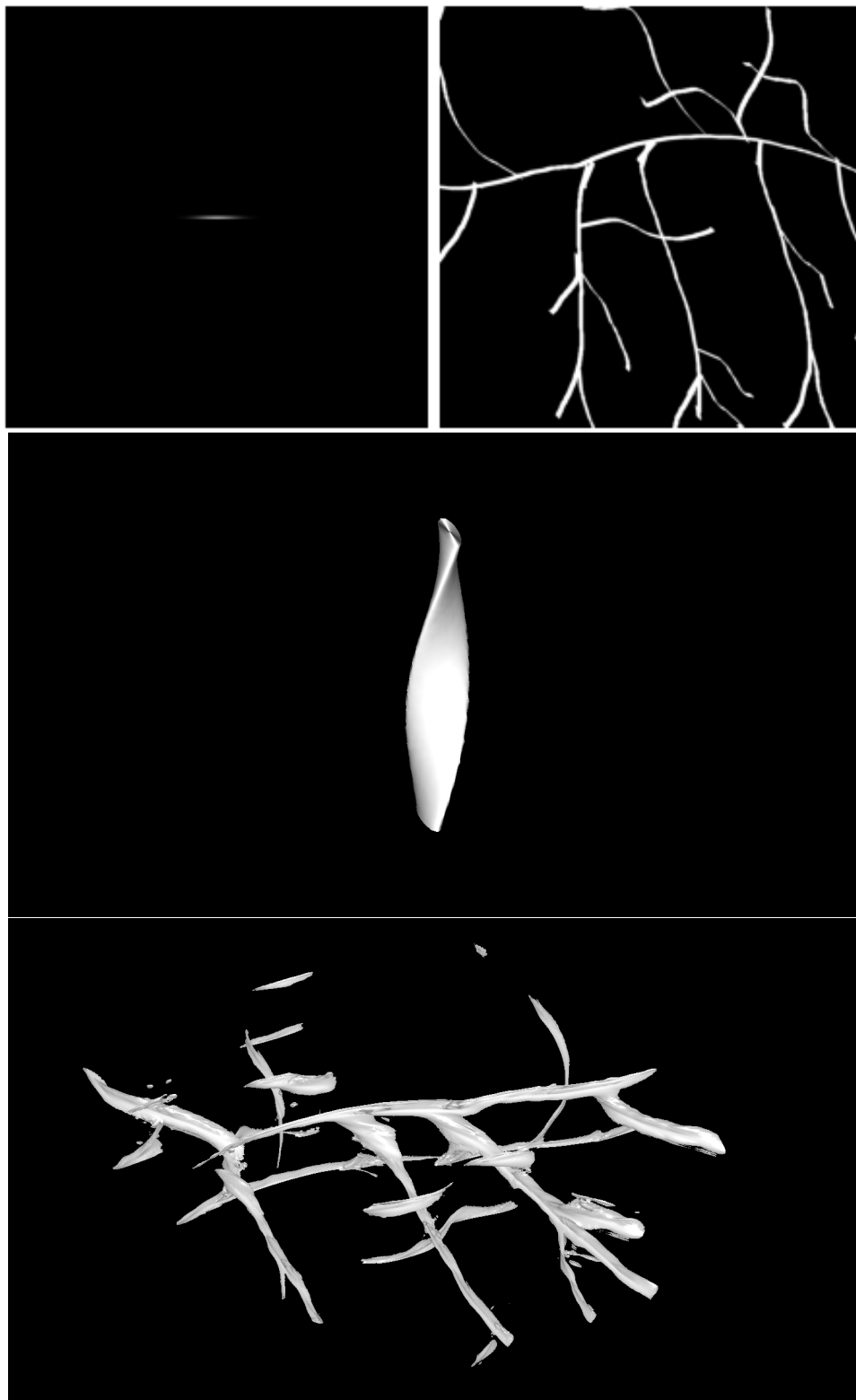


Figure 3.12: Top left : lifting kernel in the plane. Top right : volume of orientation dependent kernels. Center : example synthetic image used as training set. Bottom : Orientation-lifting of previous image with thresholding at 50% of max intensity.

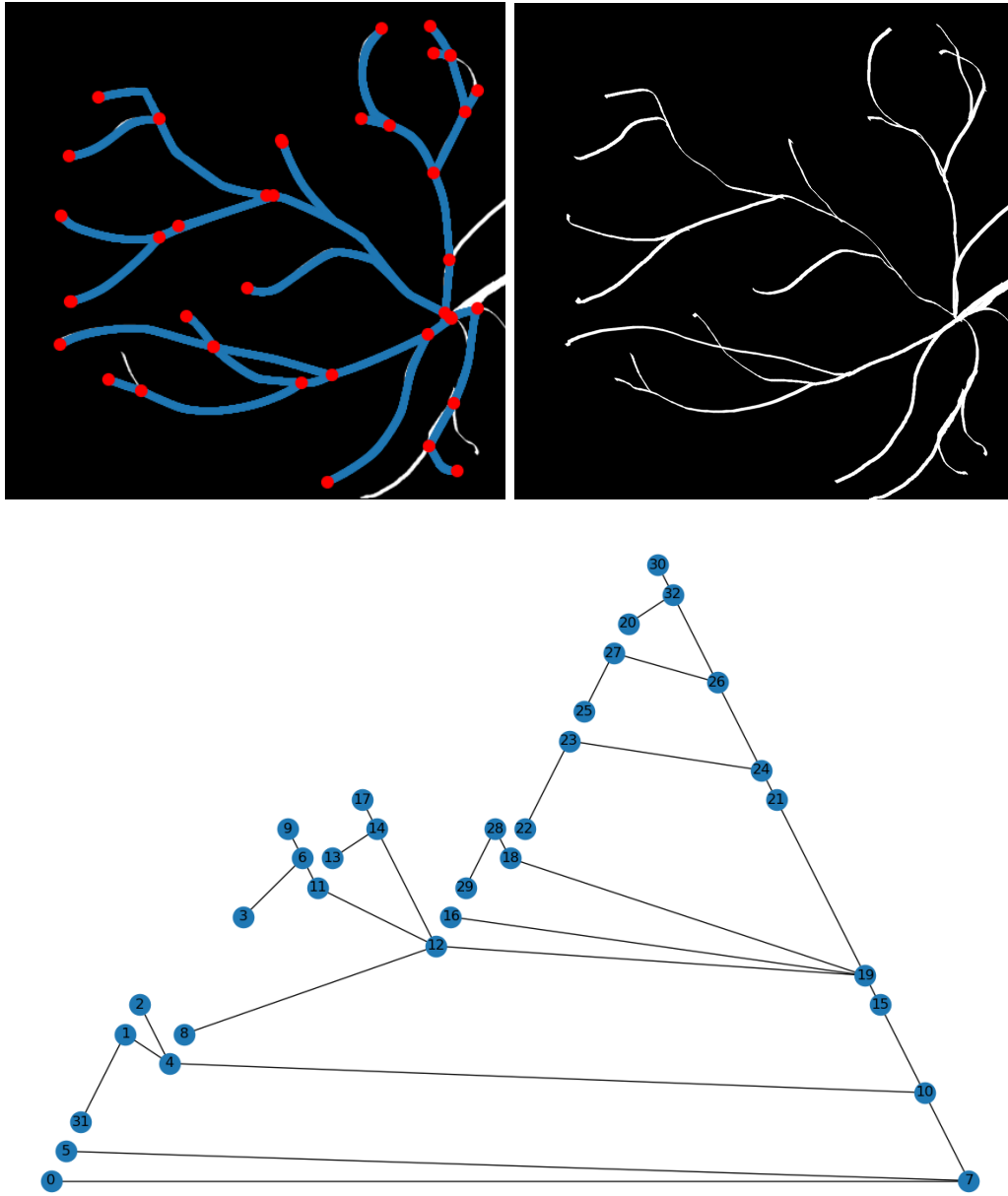


Figure 3.13: Geodesic graph on half of the synthetic validation image, with $N_\theta = 128$. Left image shows detected landmarks points and the geodesics linking them, right image shows the input image. Down is the underlying geodesic graph.

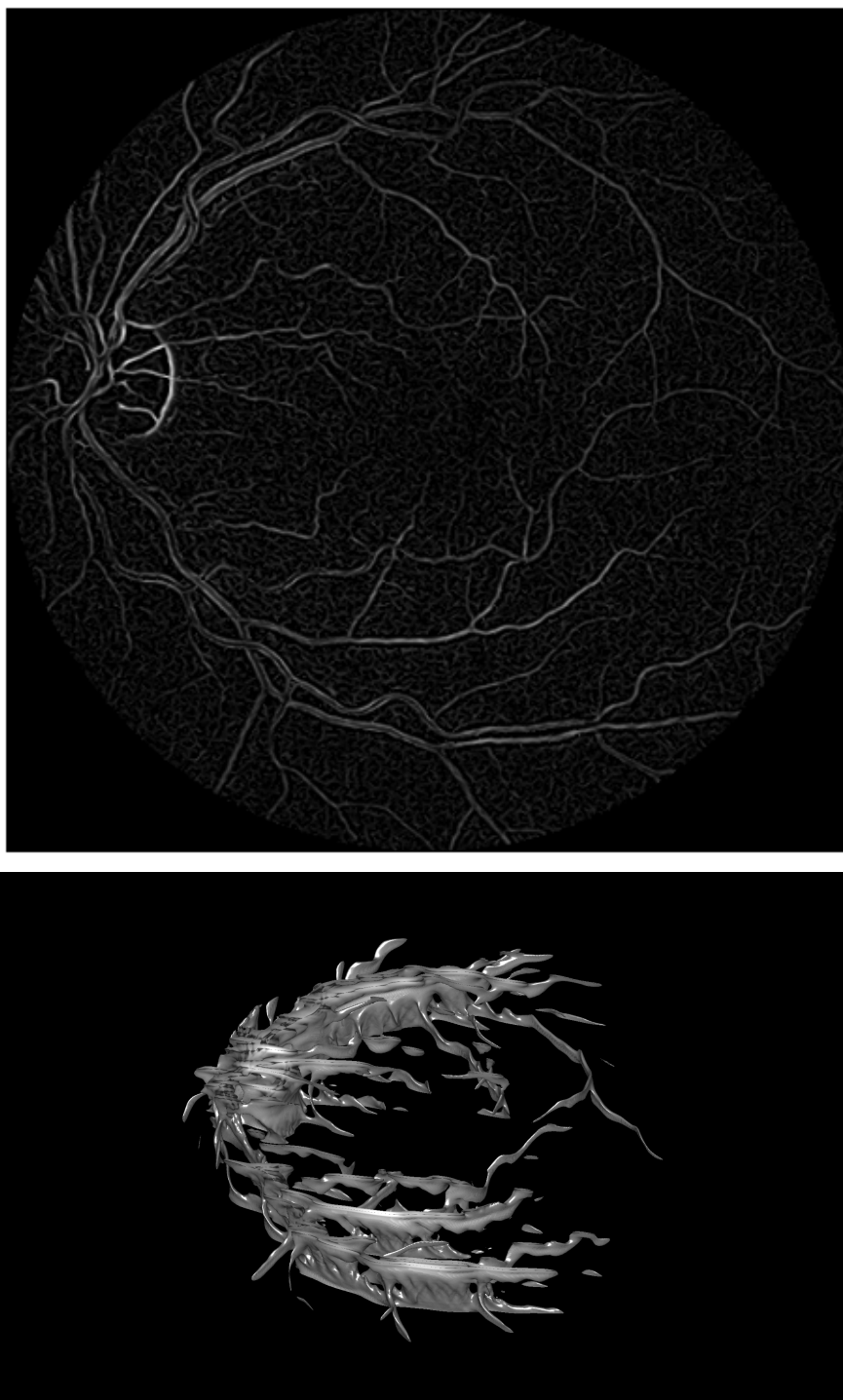


Figure 3.14: Examples of a vesselness score on IOSTAR data used to define the Riemannian metric in the distance model. Top image : a sample (same as in Figure 3.5) from IOSTAR data with Frangi filter applied. Bottom : same image lifted to the position-orientation space (only the isomap is plotted for easier visualization of the 3D volume).

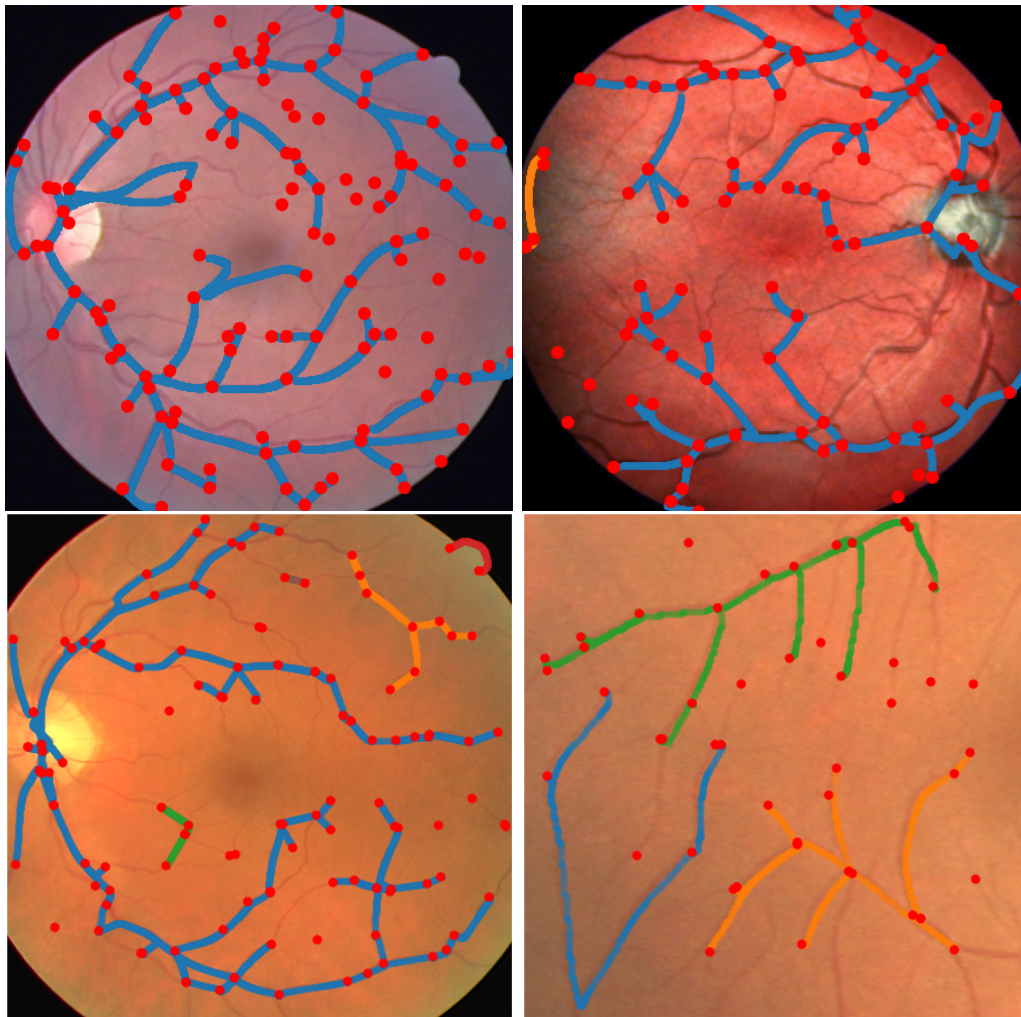


Figure 3.15: Geodesic tracking performed on four validation images from the DRIVE and IOSTAR dataset, with $N_\theta = 64$. Red points are the detected landmarks and curves are the selected tree structures.

(*crossings* and *bifurcations*) whereas we also added the additional *endpoint* class. Figure 3.15 shows a sample of our tracking performed on eye fundus data.

Rat Brain ULM data

The main ideas of the data processing for ULM data has already been described in Section 3.2.2.

We want to make full use of ULM data and use the initial set of microbubbles path from the available data Chavignon et al. 2020 to construct the cost function C in the relaxed Reeds-Shepp model as detailed in Section 3.2.3. With this goal in mind we define W by building directly the Orientation score from the histogram of microbubbles in the dataset just like it is done for the input $2D$ image, but this time we add the orientation of the given velocity vector for the orientation coordinate. After renormalization it gives us a function $W(x, \theta)$ with values between 0 and 1.

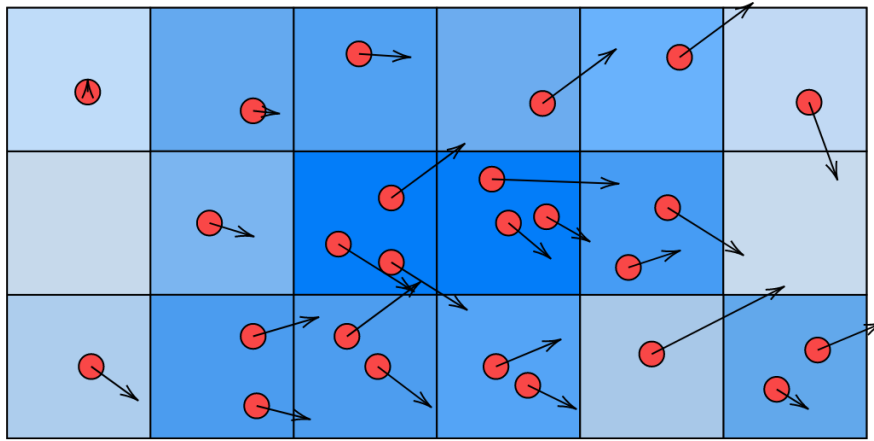


Figure 3.16: Illustration of the process to make the cost C in the position-orientation domain : red points indicate the position of bubbles, the arrows their velocity vectors and the blue grid is the pixels or voxels over which we count the bubbles. After the discretisation of the space of position and orientation we count the number of detected microbubbles in each "voxel" with explicit position from the position of the microbubble and orientation by taking the angle of the velocity.

See Figure 3.16 for an illustration of the process used in computing the vesselness score on ULM data.

Results : With the described approach to learning the detection of landmarks, we were only able to reach low mean F1 scores of around 20 % (computed on 512×512 patches). Even with such a low score on the detection-classification task, we are able to track a few of the vessels in the image, as shown in Figure 3.17. Still, some big vessels remain untracked because some points were not found at their tips.

Discussion

Application to real world data does not seem to work in a very satisfying way.

We may note the following behaviours observed after training with different hyper-parameters :

- The recovery of the geodesic tree structure is highly sensitive to change in hyper-parameters (in the definition of the metric tensor or the dependance on the position of the detected points).
- Our framework is thought for ULM images and does not necessarily adapt well to the eye fundus images dataset considered in the tracking step, although results might get better if one can tune the Orientation Score well enough such that orientation are well separated and landmarks can be linked i.e. reasonably close for the geodesic distance.
- Defining the Orientation-dependent cost function from the position of the microbubbles and their estimated velocity vector seems to be a good approach

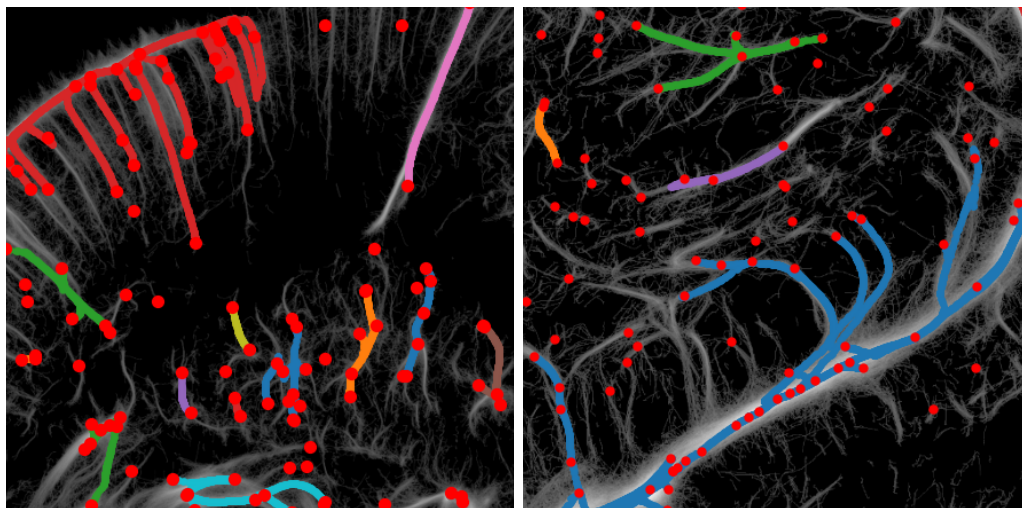


Figure 3.17: Geodesic graph on patches cropped from the ULM validation dataset (taken respectively from left and right parts of rat brain), with $N_\theta = 64$. Red points are the detected landmarks and curves are the selected tree structures.

to perform tracking on ULM data as we can see from recovered geodesics in Figure 3.17

- The results on the synthetic images tend to show that if we can provide a good enough segmentation it would be relatively easy to provide a good detection of landmarks and retrieve a good geodesic tree tracking.

3.2.5 Evaluation on segmentation score and discussion

To better understand the role of each part of our method, let's try to see what kind of result we can get, still looking at in vivo data, but in the ideal cases where we have access to the segmentation of the vascular network and the location of the landmarks. To assess the results to be expected from our approach, we will compute a F1 segmentation score that we deduce from naive and straightforward transformation of our tracking into a candidate segmentation mask. In the formation of the Riemannian metric model, what differs here is that instead of lifting a vesselness map computed directly from the input image, we use the segmentation mask available in the IOSTAR dataset which should help a lot to find the right geodesics. Samples of the results obtained on the eye fundus images is shown in Figures 3.18 and 3.19. A segmentation mask is proposed by adding a width to the computed geodesic curves forming our minimizing trees. Assessment of the result is done by computing an F1 score between the proposed segmentation and the ground truth. As we don't really want to get into the trouble of determining a complex model of the width to be taken, we simply compute many F1 scores for each image with many different widths (uniformly sampled in the log domain) and take whichever width maximizes the score each time.

To put it formally, we investigate the efficiency of our approach by evaluating a slightly modified F1 score. Indeed, here we hope to establish what kind of

performance it is reasonable to expect when comparing to classical segmentation approaches.

Let's note $F1(\hat{Y}, Y)$ the F1 similarity score computed between an attempt segmentation $\hat{Y} \in \mathbb{R}^D$ and the annotation $Y \in \mathbb{R}^D$. We will note P_s the operator converting a set of curves Γ into a segmentation mask in \mathbb{R}^D for suitable comparison to an annotation, by inflating the curves with uniform width s , and let us define $\Gamma_\sigma(X, I)$ the map associating the set of landmarks X and the input image I with the set of curves defined through the process described above, i.e. computing the pairwise distance matrix, clustering the landmarks and finding the minimal tree among each cluster. σ is the threshold needed at the clustering step.

In this subsection we will be evaluating the score

$$\max_{s \in S, \sigma \in \Sigma} F1((P_s \circ \Gamma_\sigma)(X_{th}, Y_{th}), Y_{th}),$$

with X_{th}, Y_{th} being respectively the landmarks and Segmentation mask annotations, S and Σ being sets of widths and thresholds over which we evaluate the maximum. We hope that such a score will be as fair as possible in terms of what can be achieved with the algorithm, not advantaging too much our method but still giving it its chance as it is not the same in nature as usual segmentation approaches. Another remark is that instead of applying the same width over all curves, we could have designed P to select a width corresponding to each $\gamma \in \Gamma$, but this quickly becomes very costly.

The reader may note that there are still many hyperparameters hidden in the map Γ_σ , choice of those hyperparameters is highly important especially for the definition of a good metric tensor inside the geodesic distance.

Average F1 score found on the 24 IOSTAR images of our dataset is around 63% which is to be compared with the scores easily attainable with simple CNN architectures to more complex ones of 80 to 90 %. Indeed this result is quite low but one should also keep in mind that segmentation is not really the task originally at hand. (Also one may note, if it was not clear from the start of this subsection, that here there is absolutely no learning happening.)

Figures 3.18, 3.19 and 3.20 show a few examples of execution of our method with parameters $\epsilon = 0.5, \xi = 1$ and the cost C is defined using a naive Orientation Score using anisotropic gaussian kernels. One can see that if most of the vacular tree can be recovered, shortcuts problem still happen even in the position-orientation setting which shows that improvement can be made on the definition of an appropriate cost function C .

Once we have seen the kind of Tracking results one may hope to attain with our model, we will now carry out the study to find out whether it is the lack of a good Metric model or the bad detection and/or classification of landmarks that yields mitigated tracking results.

Let it be noted that in the results reported in the following table, a small grid search on the hyper parameters of the metric model has been performed. Indeed we used the case with cost computed from input image/landmarks detected with our CNN and tried to find the best set of hyperparameters for the computation of the Cost in the metric. We performed around 100 runs of our geodesic tree selection

method, thus covering only a small part of the domain of the hyperparameters, the research was done with the Bayesian research method available on the Weights and Biases platform, Biewald 2020. We then proceeded to use the hyperparameters determined by the research to evaluate the other cases.

| $\max_{\sigma,s} F1$ | Cost from GT | Cost computed from input image |
|-------------------------------|--------------|--------------------------------|
| GT Landmarks | 64% | 58% |
| Landmarks detected with model | 61% | 56% |

Table 3.1: Summary of F1 score results

As one can see from Table 3.1 one of the main reason for our low results is the lack of a better cost in our model directly computed from the input image.

3.2.6 Partial Conclusion

In this work we have investigated the possibility to recover a complete tracking of the vessels in 2D images of vascular networks. It was done using CNN techniques from the literature to extract vascular landmarks that define the main points of interest defining the network. Our method is interesting because it fits a length-minimizing tree model to the image (using geodesics in a certain geometry to represent vessels) and thus includes both topological (tree-like structure) and geometrical (fitting geodesics) information to our tracking.

Although results on real world data are not satisfying for a complete recovery of the vasculature, we have shown the potential of using ULM data and the information they carry can be used to accurately track vessels.

Further research prospects include incorporating scale information or scale equivariance to distinguish vessels and help the localization process and also provide width information

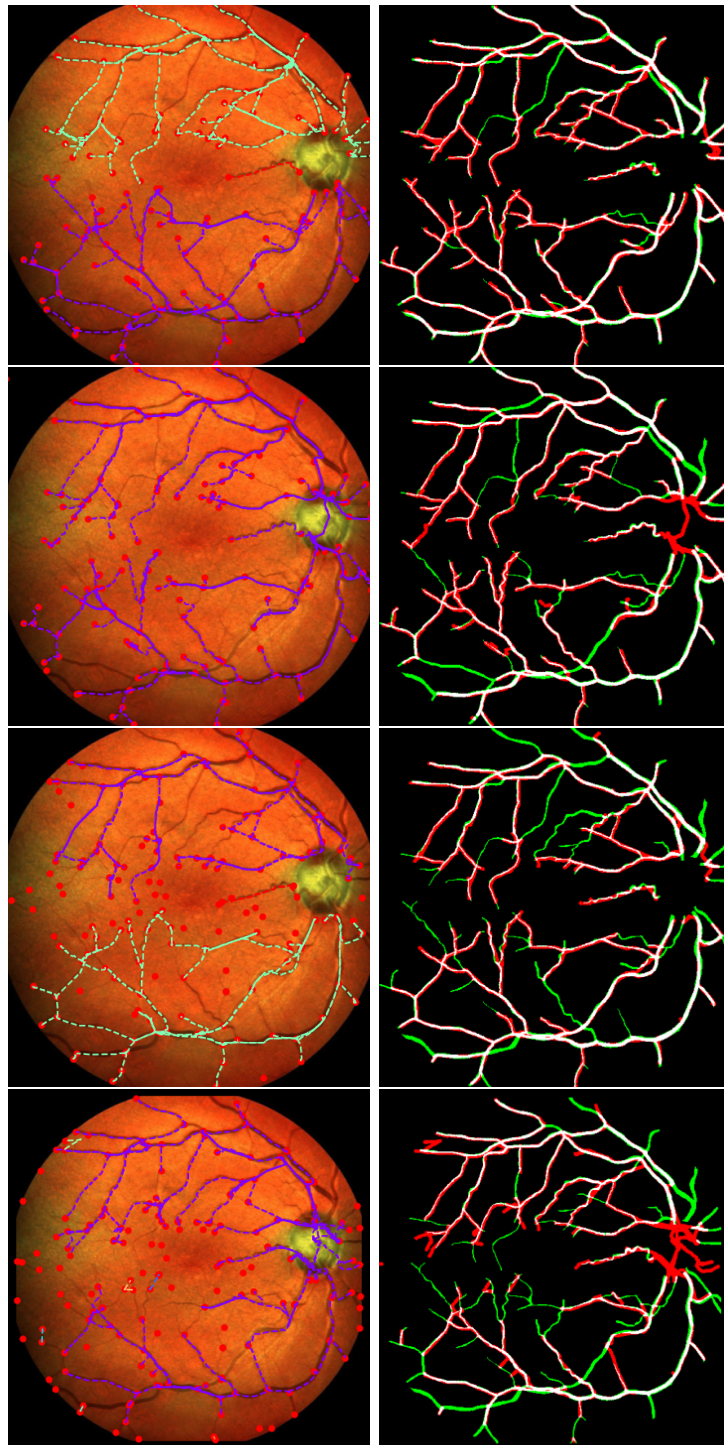


Figure 3.18: Examples of our method applied on the IOSTAR dataset. Left column : selected curves superimposed with the input image. Right column : curves projected on the grid with width, red region is the proposed segmentation, green region is ground truth, and white region is the intersection. 1st row : metric built from GT/GT landmarks. 2nd row : metric built via input image/GT landmarks. 3rd row : metric built via GT/model landmarks. 4th row : metric built via input image/model landmarks.

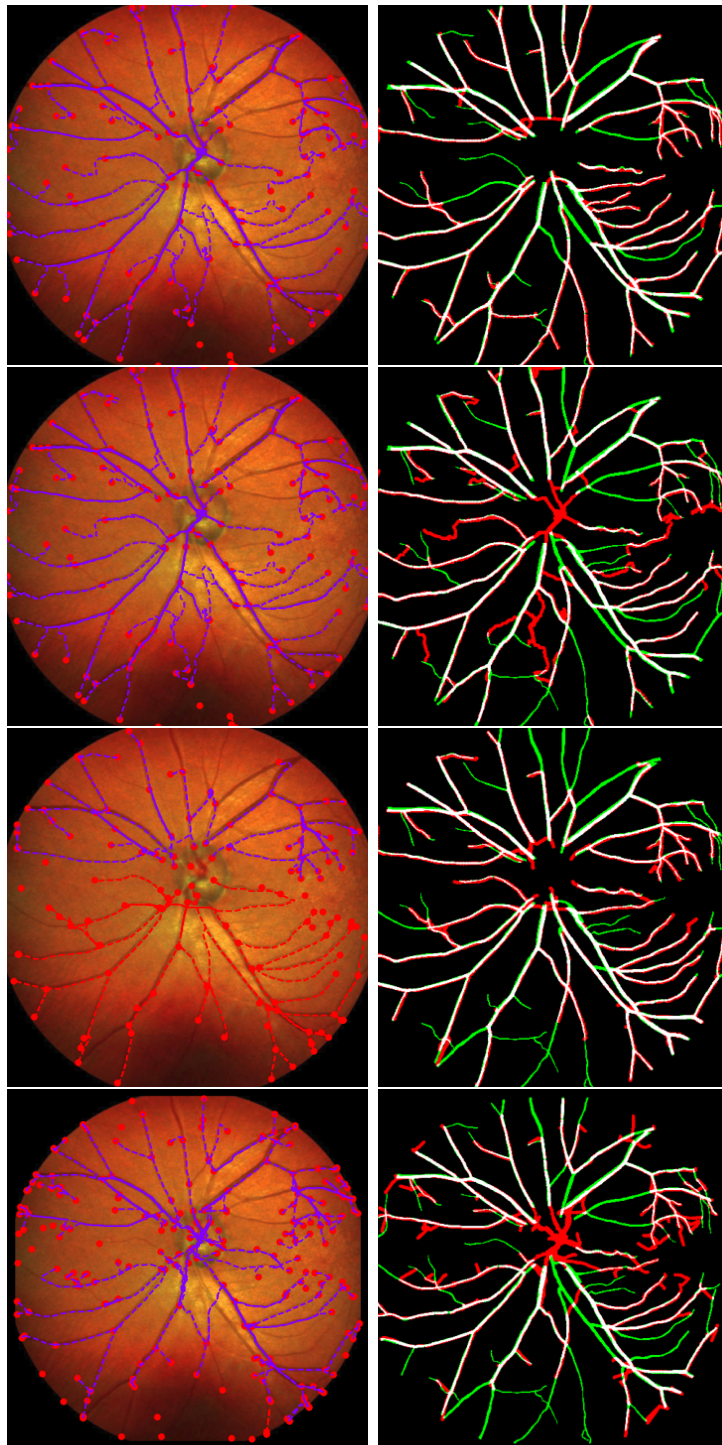


Figure 3.19: Examples of our method applied on the IOSTAR dataset. Left column : selected curves superimposed with the input image. Right column : curves projected on the grid with width, red region is the proposed segmentation, green region is ground truth, and white region is the intersection. 1st row : metric built from GT/GT landmarks. 2nd row : metric built via input image/GT landmarks. 3rd row : metric built via GT/model landmarks. 4th row : metric built via input image/model landmarks.

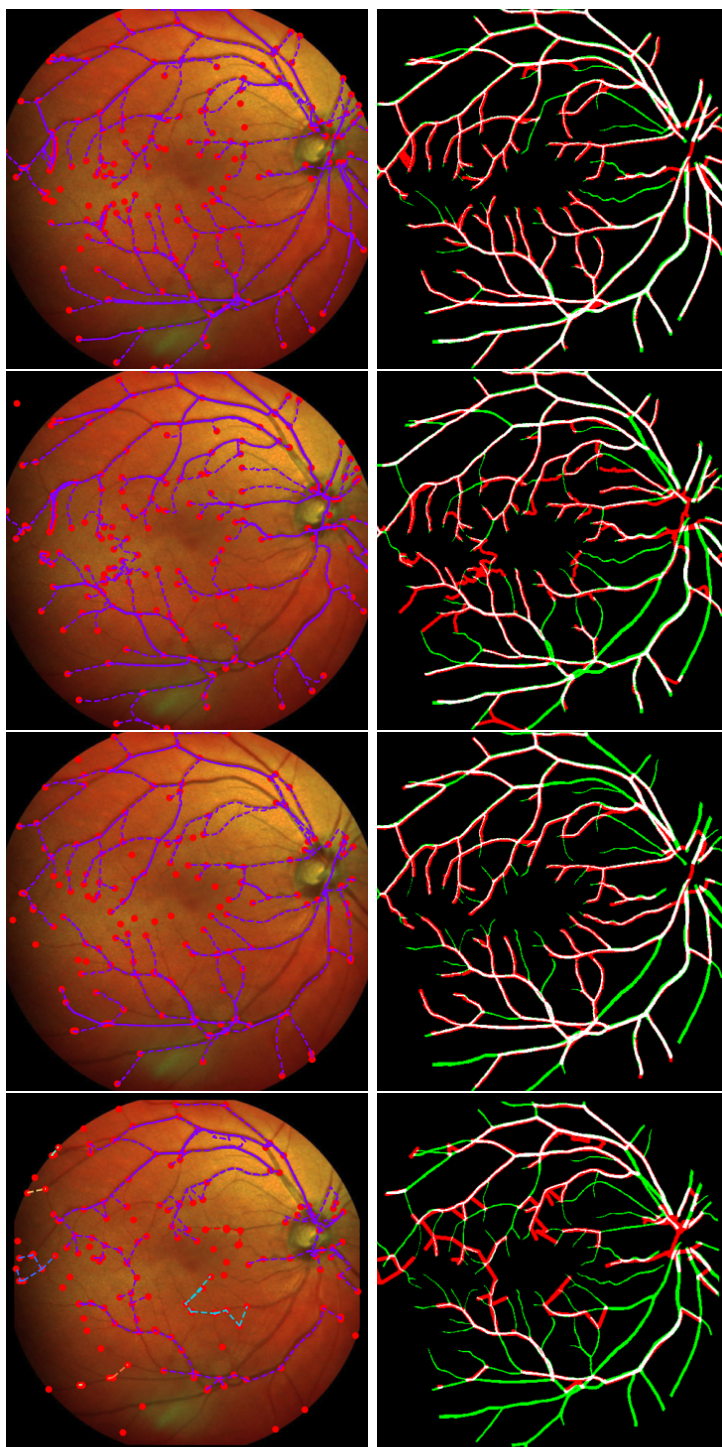


Figure 3.20: Examples of our method applied on the IOSTAR dataset. Left column : selected curves superimposed with the input image. Right column : curves projected on the grid with width, red region is the proposed segmentation, green region is ground truth, and white region is the intersection. 1st row : metric built from GT/GT landmarks. 2nd row : metric built via input image/GT landmarks. 3rd row : metric built via GT/model landmarks. 4th row : metric built via input image/model landmarks.

3.3 Riemannian metric learning

This work is a collaboration with Nicolas Makaroff and Laurent Cohen and was published first in the proceedings of the conference SSVM 2023.

Geodesic curves and distances have been used to convey geometric properties in many different applications. The usual approach of those methods is to rely on prior knowledge of the task at hand to build a Riemannian metric g explicitly from data.

The approach presented in this work tries to get rid of the bias introduced in the choice of a metric tensor by generating it from data via a Neural Network architecture whose parameters were previously optimized in a supervised learning approach with training data. Introducing such a bias is not a bad thing in itself, however, it requires an arbitrary decision from a user and parameter tuning, two issues that can be avoided by learning to generate a metric from data.

To demonstrate the effectiveness of this framework, we apply it to a segmentation task using a brain tumour MRI images dataset. By using our proposed method, we can obtain accurate results compared to traditional approaches, highlighting the capabilities of this approach. Furthermore, we also observe that our method has a remarkable ability to learn from data and somewhat generalize to unseen data.

The approach presented here is simply to recover the region to be segmented as a sublevel set of a distance function defined with respect to a Riemannian metric, or rather a unit geodesic ball. Extensive literature on the subject allows us to compute efficiently both the distance field on the grid from a set of points and to provide (sub)gradients to perform the optimization. The question of which points are taken as centers of the geodesic ball is also non-trivial and we will treat this problem by making the network to be optimized find them himself. Figure 3.21 shows examples of distance functions and their associated unit geodesic ball.

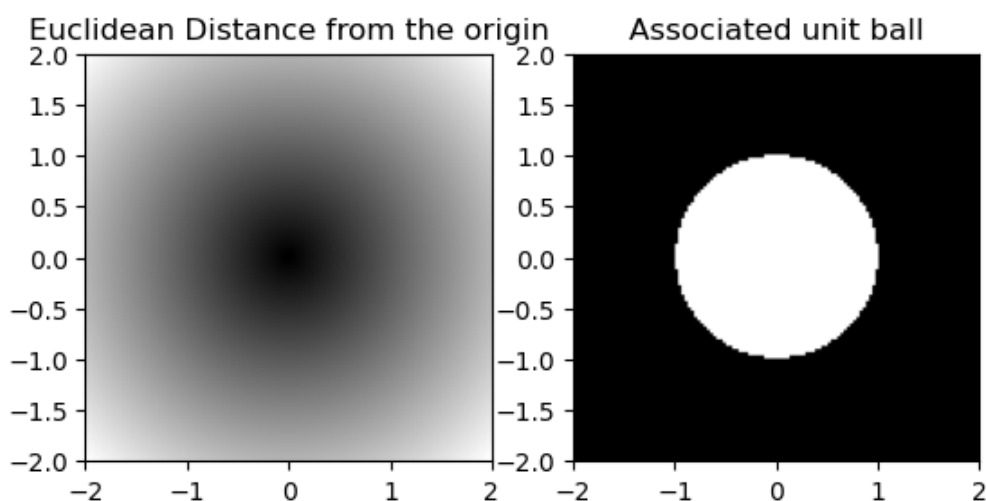


Figure 3.21: Examples of geodesic balls.

The method introduced in this work offers a powerful and flexible way of using

geodesic curves and distances in a wide range of applications in a holistic learning framework.

The rest of this section is organised as follows. In Section 3.3.1 we present the computation of geodesic distances and their gradient. In Section 3.3.2 we present the main results of our experiments and provide a discussion around our work.

Related Works

The use of geodesic distances in segmentation tasks has a long history. To the authors' knowledge, the first article to segment an image's region using a minimal path distance and fast marching is Malladi and J. Sethian 1998, with application on a 3D brain image. In the case of the segmentation of tubular tasks we can refer to D. Chen and L. D. Cohen 2016 for instance, a method that segments the 3D vascular tree by propagating the front of the minimal path distance computation. Similarly, L. D. Cohen and Deschamps 2007 segments vascular structures by introducing an anisotropic metric, determined dynamically by evaluating local orientation scores during the Fast Marching computations. Those 3 articles already use the level sets of the geodesic distance (or "geodesic balls") to provide the segmentation mask. We may also mention Benmansour and L. D. Cohen 2009b that uses geodesic curves in an higher dimensional space to track vessels (as curves with an additional width component). These works generally aren't interested in treating the task in an holistic manner and focus on providing a good model for the structures to segment, whereas this work tries to treat the problem end-to-end and generalize to a large dataset of input images.

Only a few previous methods are interested in learning a metric from data. We may mention recent works such as Scarvelis and Solomon 2022 and Heitz et al. 2021 that try to find metric tensors that fit spatio-temporal data in order to capture the velocity fields and underlying geometry of the data. The first paper is modelling trajectories as the solutions of a dynamical system generated by a Neural Network and also taking into account the dynamics of the whole population by penalizing an optimal transport cost between two consecutive timestamps. However Heitz et al. 2021 tries to interpolate a sequence of histograms with Wasserstein barycenters by optimizing over the metric tensor appearing in the ground cost. Also, there are important links between the Wasserstein optimal transport, its dynamical formulation and geodesics, for further reading, we refer to Ambrosio, Brué, et al. 2021. These works propose interesting frameworks to work with, but they are not focused on generalizing the generation of the metric tensors.

As previously seen in Chapter 2, Benmansour, Carlier, et al. 2010 laid the ground for the differentiation of the geodesic distance with respect to the metric in the Fast Marching algorithm. They then proceed to apply it in the setting of inverse problems to retrieve the metric from distance measurements. Its only concerns were to solve inverse problems involving the geodesic distance, whereas we go one step further by including a Fast Marching module in a deep learning segmentation procedure. The sub-gradient marching algorithm is briefly described in section 2 as it is essential to our framework to propagate through the Fast Marching module and carry the learning step.

In terms of Deep Learning, we might add a few references such as the classical

Ronneberger et al. 2015 and He et al. 2015 that respectively introduce the UNet and ResNet architectures, which are used for our method and as baseline comparisons. For a review of deep learning methods in medical imaging one might refer to Zhou et al. 2021. The very general methods directly producing segmentation from medical images are already quite efficient, but they suffer from a lack of robustness and do not impose a lot of structure on the segmentation that comes out of the network. Contrary to this, our work allows us to impose a lot of constraint on the topology of the segmented region (namely a set with trivial topology).

3.3.1 Model

The proposed method presented in this study uses a neural network, specifically a modified version of the UNet architecture, to segment regions of an image as geodesic balls with respect to a metric. The metric is obtained by training a convolutional neural network (CNN) to provide both the metric and the center or seed of the geodesic ball. The framework, as shown in Figure 3.22, processes the input image using the encoder component of the UNet, resulting in a vector representation of the image. This vector is then passed through two separate decoders to perform distinct tasks.

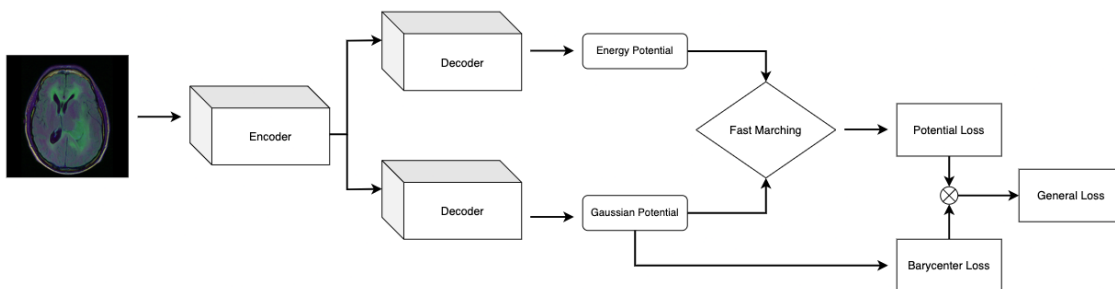


Figure 3.22: Diagram of the framework from the input image to the loss.

The first decoder predicts the potential ϕ to be used by the fast marching module, which can be computed using the HFM library. The second decoder predicts a Gaussian potential that represents the probability of the presence of the region's barycenter in a given area, which is also provided as a seed to the fast marching module. The distance map generated by the fast marching procedure is then used to find a geodesic ball for segmentation. The expected segmentation is compared to the predicted segmentation, and the theoretical barycenter is compared to the predicted Gaussian potential to compute the error.

The distance computation module can be written as a function of both seed points and input metric. The metric ϕ is defined as the output of a CNN architecture, such as the widely used UNet, with θ being in the space of parameters. We enforce positive and non-zero properties of the metric by taking $\phi = f_{\theta}(u)^2 + \epsilon$, with u being the input image and f_{θ} being a CNN, with $\theta \in \Theta$ and Θ the space of parameters. To avoid solutions that distribute a lot of mass everywhere, as noted in Benmansour, Carlier, et al. 2010, we ensure that the total mass of the metric is reasonable by applying a transformation $\phi \mapsto \frac{\phi}{\max(\frac{1}{\lambda} \|\phi\|_{1,1})}$ that bounds from above

the L^1 norm at a fixed level λ (We chose in this work to empirically bound the total mass at 5).

UNet

In this study, we focus on the task of potential generation and employ two different architectures commonly used for image segmentation: the UNet Ronneberger et al. 2015 and a combination of the UNet and ResNet He et al. 2015.

The depth of CNNs can cause the problem of vanishing gradients, which can affect model performance. To address this, we propose the use of ResNet-UNet, a combination of the UNet and ResNet-34 model in the encoder portion of the network. ResNet-34 benefits from deep residual learning and comprises of a 7x7 convolutional layer, a max pooling layer, and 16 residual blocks.

By combining these architectures, ResNet-UNet can capture fine and coarse features of input images and learn deeper and more complex representations. This results in a more accurate and robust model for image segmentation tasks, as demonstrated by our experimental results. Additionally, we introduced modifications to the expansive path of both networks, implementing a dual expansive path system to predict potential energy and a Gaussian potential for the prediction of barycenter. These modifications are illustrated in Figure 3.22. Overall, our proposed model demonstrates promising results for potential generation tasks.

Generating masks with geodesic balls

Applications may take advantage of topological priors on the label to reconstruct. For instance one may need to recover regions in an image that we know to be path-connected and of trivial topology. Such regions might be modelled as balls related to a specific distance and recovered as indicator function of such a ball. Formally, we expect for a set E to recover an indicator function as $\chi_{d_\phi(x_0, \cdot) \leq 1}$ for well chosen $x_0 \in \mathbb{R}^d$ and $\phi \in L^1(\Omega)$.

With this method of building masks for specific tasks, we can try to generalize using a neural network architecture and find good potential ϕ to segment interesting regions in images. To do this we would need to compute the gradient of a chosen loss function and thus would need to differentiate the mask, that is why we will replace the indicator function on the unit ball, that would yield zero gradients almost everywhere, by a sigmoid that will smoothly interpolate between the value 1 in the region inside the unit ball and 0 outside. Given the distance map $d_\phi(x_0, \cdot)$, our mask then becomes

$$\chi^\delta(d_\phi(x_0, \cdot)) = 1 - \frac{1}{1 + \exp(-(d_\phi(x_0, \cdot) - 1)/\delta)},$$

which approaches the characteristic function of the unit ball as the parameter δ approaches 0. δ will be taken typically of the order of the size of pixel, i.e. approximately the inverse of the image size.

Figure 3.23 shows how it is possible to approach the characteristic function of different sets with this formulation. This problem is not convex, so solutions may vary depending on the initialization for instance, but it seems that most of the time

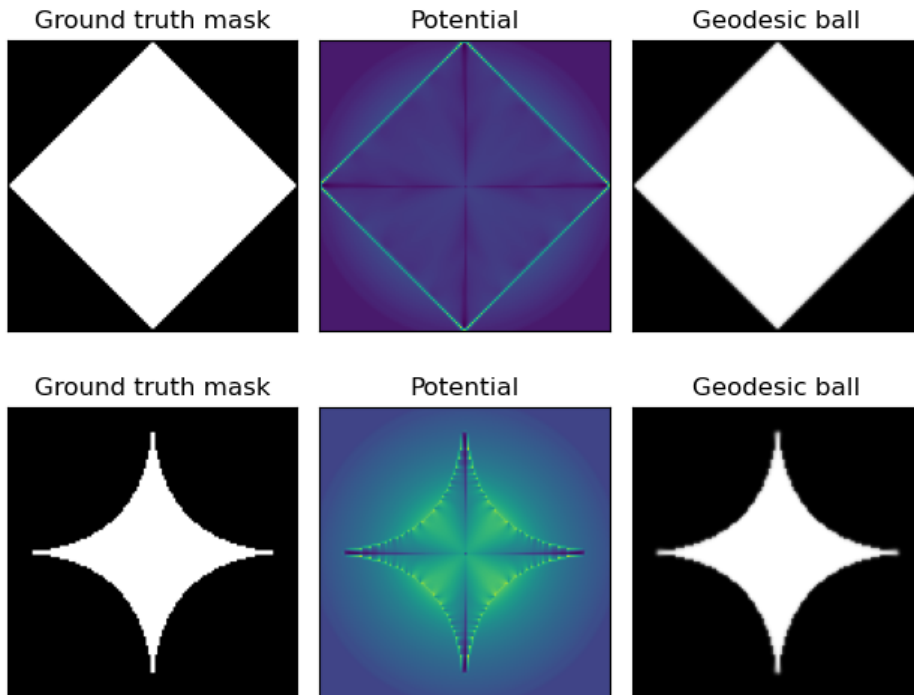


Figure 3.23: Example of recovery of an isotropic metric fitting two regions by minimizing $\|\chi^\delta \circ d_{\phi^2} - y\|_2^2$ with respect to ϕ , where y is the ground truth mask, $\delta = 0.01$. x_0 is taken as the center of the mask to be recovered.

potentials converge to a solution that puts a lot of mass on the edges of the mask to recover. The seed here is fixed to x_0 the center of the balls to be fitted, and the potential ϕ is directly optimized using automatic differentiation and ADAM with a "learning rate" equal to 0.01. ϕ^2 is taken as input for the fast marching algorithm instead of ϕ as an easy way to smoothly enforce positivity of the potential.

3.3.2 Experiments

As announced before, our experiments were led on tumour segmentation task.

Data

To conduct our experiments we have used a dataset of Brain MRI segmentation task that is the TCGA_LGG database openly available on the internet Pedano et al. 2016. This database contains MRI scans of patients with brain tumours. They correspond to 110 patients (resulting in 1189 images) included in The Cancer Genome Atlas (TCGA) lower-grade glioma collection with at least fluid-attenuated inversion recovery (FLAIR) sequence and genomic cluster data available. We removed tumour with multiple connected components. This dataset is composed of the data of 110 patients. We have used the set of 2D MRI images as our learning and training datasets. We have set aside 5 patients' data to form a test set as independent as possible (whereas two images from the same patient can be separated in the training and validation set, test data are always the result of a different acquisition from

the training and validation set). We applied data augmentation on the training images to increase the diversity of the training set and improve the generalization of the model. The data augmentation techniques used were: horizontal flipping with probability $p=0.5$, vertical flipping with probability $p=0.5$, random 90-degree rotation with probability $p=0.5$, transpose with probability $p=0.5$, and a combination of shifting, scaling, and rotating with probability $p=0.25$. We respectively set the shift limit, scale limit and rotation limit to 0.01, 0.04, and 0 (as we already perform rotation). We computed the tumour seed using a Euclidean barycenter of the mask region.

Model Training Procedures

The UNet architecture was employed for the task of image segmentation in this study. The model was initialized with Kaiming distribution and trained using the Adam optimizer, which has been widely used in literature due to its capability to adjust the learning rate during training. The learning rate was set to 1e-3, which is a commonly used value in CNNs, as it provides a balance between achieving convergence and avoiding overshooting the optimal solution. In order to optimize the model's performance, to penalize the error between the prediction mask and the groundtruth mask we used a combination of Dice loss and Binary Cross-Entropy (BCE) loss (Equation (3.6)).

$$\mathcal{L}_S(x, y) = \frac{2 \times \sum_{i=1}^N x_i y_i}{\sum_{i=1}^N x_i + \sum_{i=1}^N y_i} + \frac{1}{N} \sum_{i=1}^N (y_i \log(x_i) + (1 - y_i) \log(1 - x_i)) \quad (3.6)$$

To control the error on the seed prediction a Binary Cross-entropy \mathcal{L}_H loss was used.

The final loss is:

$$\mathcal{L}(x, y, h^1, h^2) = \mathcal{L}_S(x, y) + \mathcal{L}_H(h^1, h^2) \quad (3.7)$$

The Dice loss function, which is known for its ability to handle imbalanced data, was combined with the BCE loss function, which provides stability during training.

In order to determine the distance between two barycenters, a transformation of the position coordinates into a Gaussian potential is used, based on the following formulation:

$$f(x, y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - b_1)^2 + (y - b_2)^2}{2\sigma^2}\right) \quad (3.8)$$

Here, (b_1, b_2) represent the coordinates of the barycenter. At inference time, the predicted potential is used to identify the maximum location, from which the barycenter coordinates can be extracted.

The model's architecture was initialized with 64 feature maps, which has been shown to be a suitable number for high resolution images, and a batch size of 16 was used during the training process. This combination of hyperparameters allowed the model to effectively use detailed information from the input image while maintaining a balance between generalization and overfitting, as demonstrated by the results presented in this paper. Perhaps it should be clarified that since the two decoders

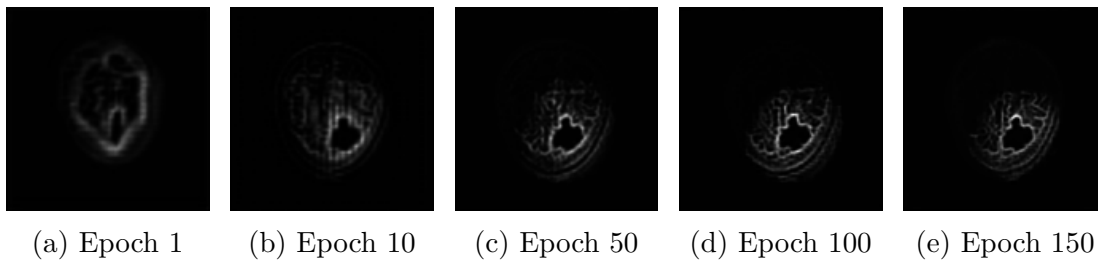


Figure 3.24: Evolution of the predicted potential taken as input in the Fast Marching Module.

are different and predict two different things, these new parameters do not assist the segmentation compared to the direct method.

Potential Analysis

The potential generated by the neural network was analyzed with respect to the number of training epochs. Results show on Figure 3.24 that the output distribution quickly converged towards the boundaries of the tumour to be segmented. However, as training progressed, the contour of the tumour sharpened and boundaries became more distinct and at the same time we can see the brain edges removed. The potential in the end only holds detailed information of the contours in a small area around the tumour.

3.3.3 Segmentation Experiments

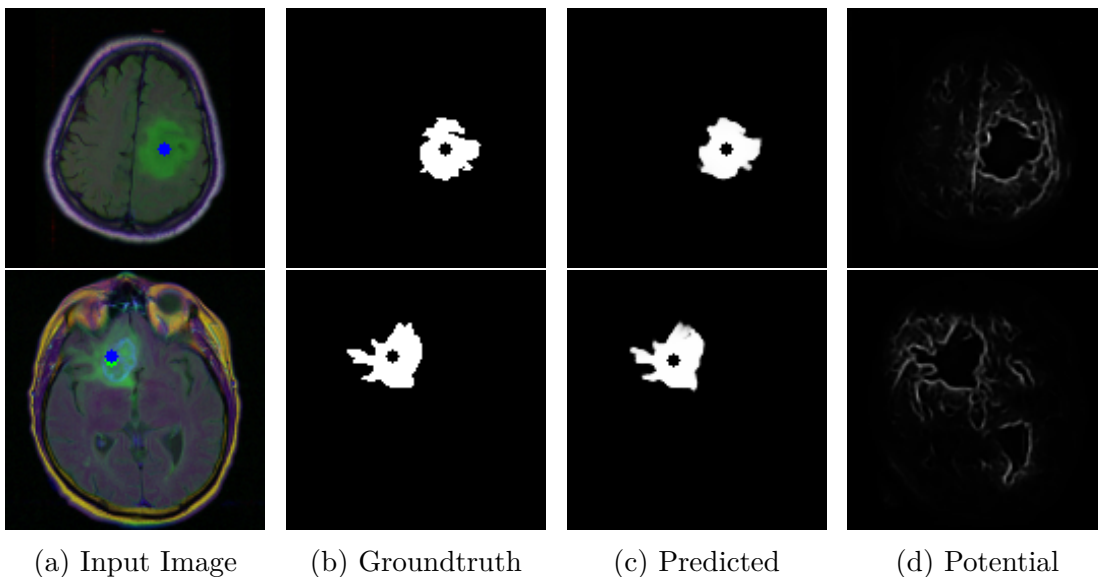


Figure 3.25: Results of the segmentation on validation data. On the input image, the blue and green dots are respectively the groundtruth and predicted seed.

We compared our method to a standard UNet segmentation approach. As can be seen in the results plotted in Figure 3.25, our method demonstrates clear edge

Table 3.2: Segmentation results (IOU) on the TGCA_LGG brain MRI database.

| Name | Dice | IOU | Hausdorff | FPR | FNR |
|-----------------------|-------|-------|-----------|-------|-------|
| UNet | 0.862 | 0.869 | 2.313 | 0.007 | 0.05 |
| ResNet UNet | 0.873 | 0.877 | 2.257 | 0.006 | 0.07 |
| FM UNet (ours) | 0.825 | 0.823 | 2.505 | 0.011 | 0.064 |
| FM Resnet UNet (ours) | 0.863 | 0.866 | 2.248 | 0.009 | 0.04 |

detection. The well-defined contours produced by our method are a result of its ability to take into account the morphology of the image, which traditional filters are not able to do. Furthermore, the problem-specific nature of our method allows for improved performance in image segmentation. Classical metrics allows us to compare quantitatively the results of our segmentation. Overall we recover the same precision on the segmentation mask with minimal improvements of the symmetric Hausdorff distance. However the convergence towards an acceptable solution is faster when combined with the Fast Marching Module since with only a approximate potential the method converge to a relatively close segmentation. Time gives the neural network to more precisely learn the filter and sharpens the edge of the tumour. A general observation from the segmentation in Figure 3.25 is that the method when failing to predict correctly a pixel tends to create false positive rather than true false. The Table 3.2 shows how our method has a high recall controlling that there is a very low number of false negative. We performed the training with the library *HFM* and the heat method and recorded same results. Overall the UNet architecture shows difficulties to precisely learn the potential while from a metric point of view the ResNet-UNet performs comparatively as the classical segmentation technique using CNNs.

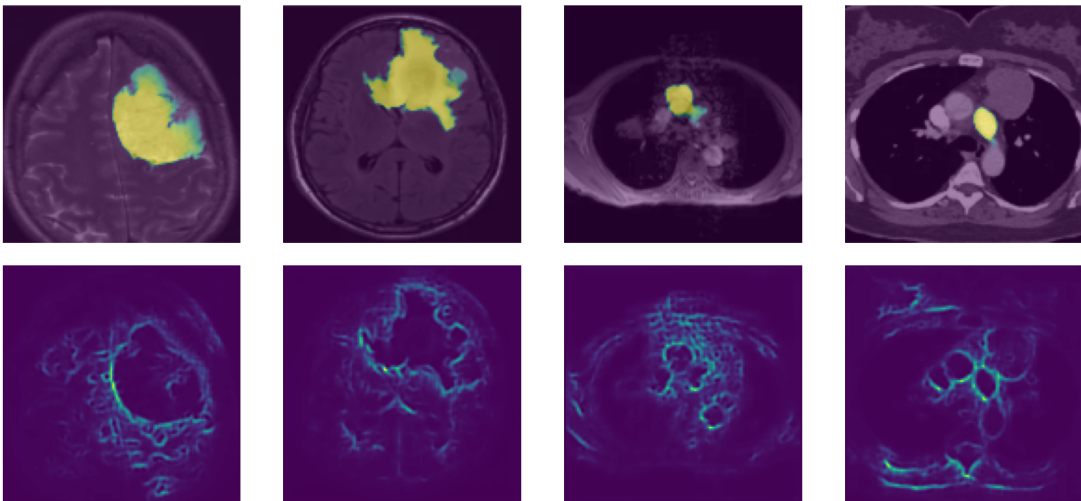


Figure 3.26: Results of the Fast Marching Energy CNN for images outside the scope of the training database. Top row: segmentation of outside the training scope. Bottom row: Potential output by the CNN before fast marching.

We further studied the properties of the generated potential of our CNN by testing it with dissimilar MRI images found randomly through an image search

on Figure 3.26 where activated areas correspond to the segmentation ranging from yellow to green for confidence. The results for the last two MRI images show that while the algorithm does not properly segment the tumour (as the predicted barycenter for initialization of the Fast Marching is not correctly placed), the learned filter detects small contours similar to tumours, focusing on the shape of the different objects.

3.3.4 More examples

In Figure 3.27 and Figure 3.28 we show images where our trained model achieves its best and worst performances for the F1 score.

One might note that the best scores seem to be obtained when the target mask is large and looks like a simple ball.

In the worst cases, usually our model fails because the zone to be segmented is very narrow and/or it fails to provide a good seed inside the zone to be segmented.

Figure 3.29 shows that for a large part of our dataset we achieve high F1 scores and for a small part of it the score is very low. This analysis is consistent with what is shown in the worst cases in Figure 3.28 where the scores achieved in the worst cases are basically 0 because we completely fail to locate a good seed point from which to compute the geodesic distance. In fact, contrary to what one would expect, it is almost easier to learn to provide good potential for the segmentation than to accurately predict the barycenter of the target segmentation mask as we are trying to do here (which is essentially a projection of the mask in \mathbb{R}^2 , a very low dimension space. We also compare two different approaches to selecting a seed between choosing the mean predicted value and the maximum probability value, and as one can see, the second approach reduces the proportion of images where our trained model performs very badly (the bin indicating almost 0 score goes down from $\sim 7.5\%$ to around 0.25%), and the overall performance on the dataset is also higher (83% vs 85%).

For the sake of completeness we also show a few samples of our result on the test set only in Figures 3.30 and 3.31.

And also the density of the F1 score over the test set in Figure 3.32

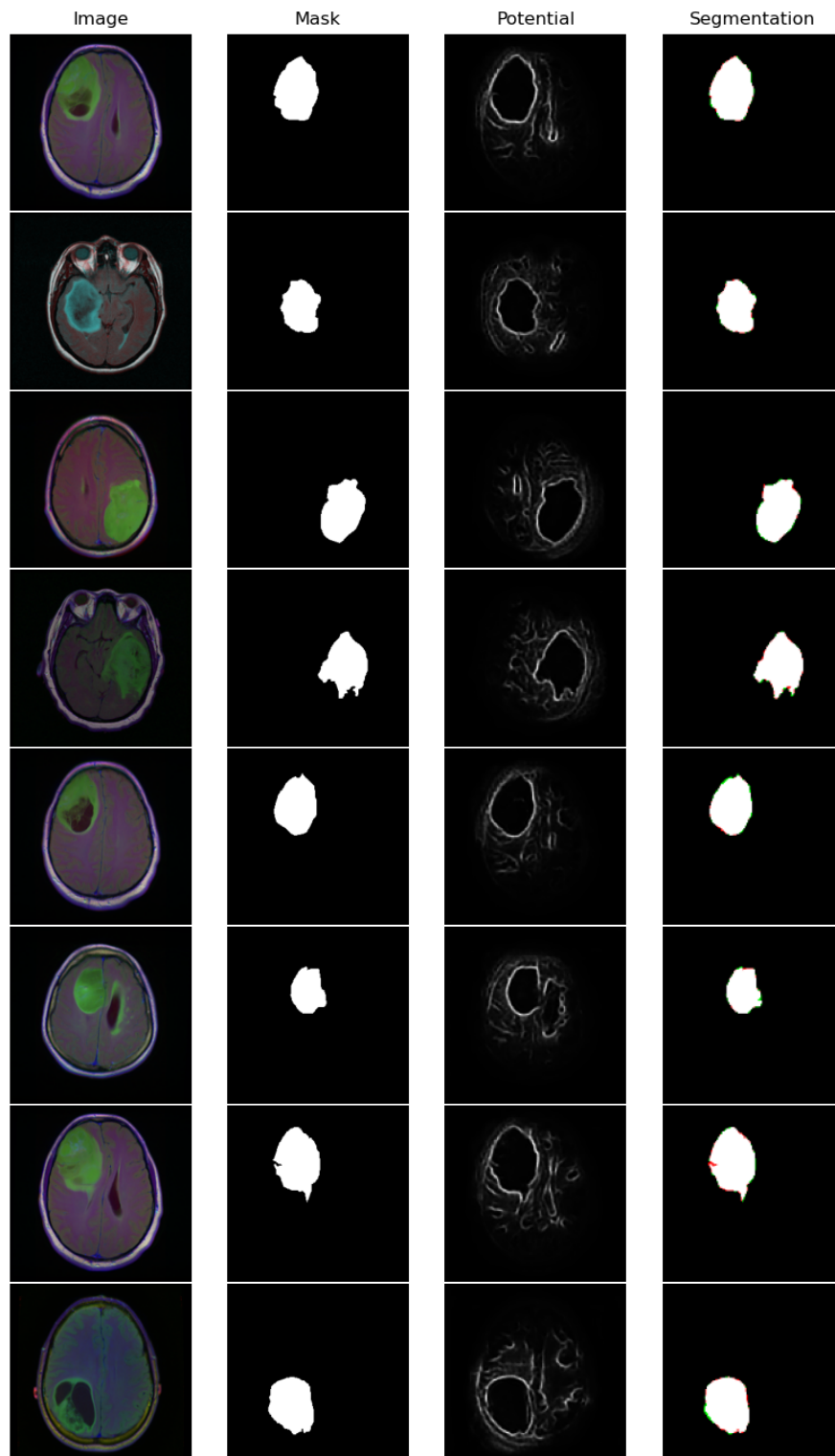


Figure 3.27: Examples where our FMECNN model achieves its highest scores. From Left to Right, figures show the input image, the target segmentation mask, the potential computed by our trained model, and the proposed segmentation (superposed with the target, red canal is the proposed, green canal is the target, and blue is the intersection).

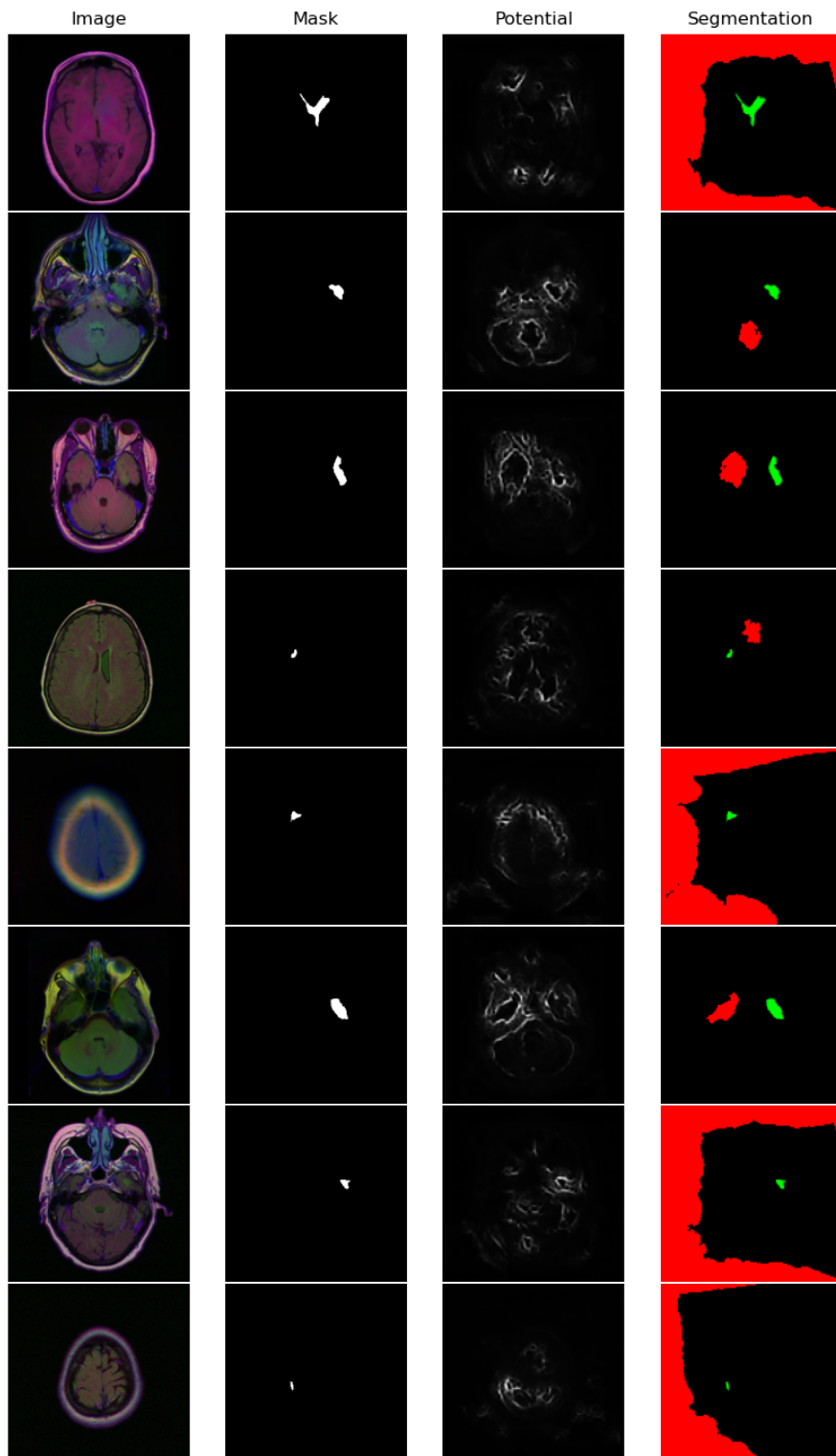


Figure 3.28: Examples where our FMECNN model achieves its lowest scores. From Left to Right, figures show the input image, the target segmentation mask, the potential computed by our trained model, and the proposed segmentation (superposed with the target, red canal is the proposed, green canal is the target, and blue is the intersection).

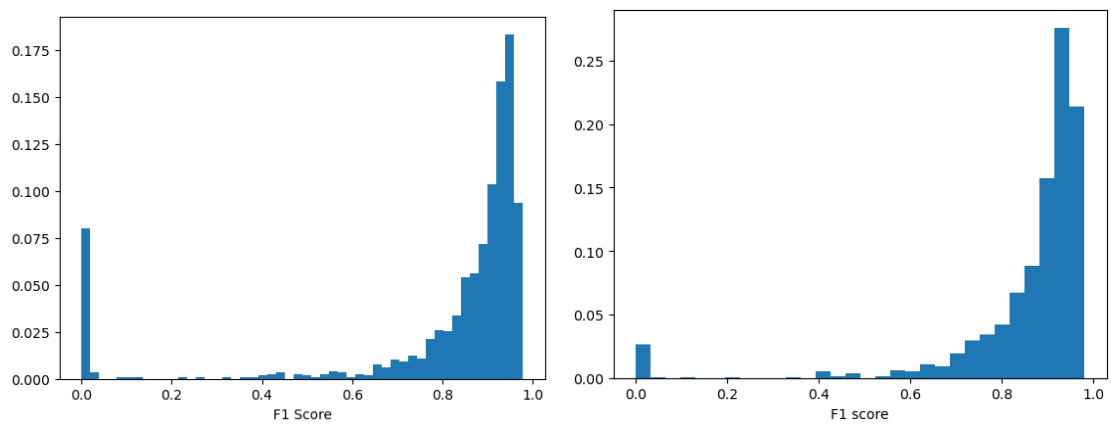


Figure 3.29: Density plot of the F1 score of our model over the whole dataset (training + validation + test set). Left : the center of the ball is selected as the mean of the predicted position of the barycenter of the segmentation. Right : the center is taken as the maximum value point of the prediction for the position of the barycenter.

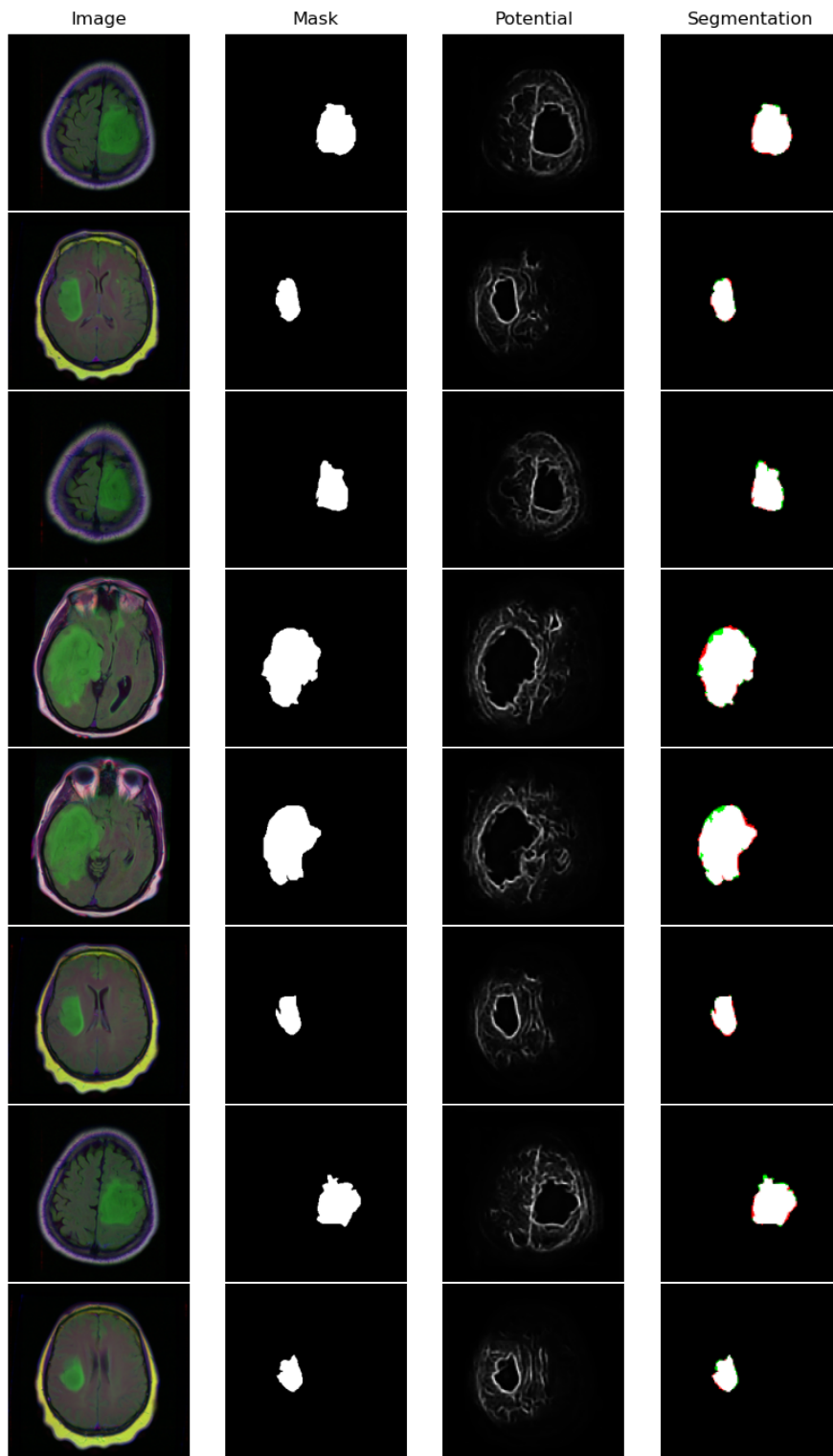


Figure 3.30: Examples where our FMECNN model achieves its highest scores. From Left to Right, figures show the input image, the target segmentation mask, the potential computed by our trained model, and the proposed segmentation (superposed with the target, red canal is the proposed, green canal is the target, and blue is the intersection).

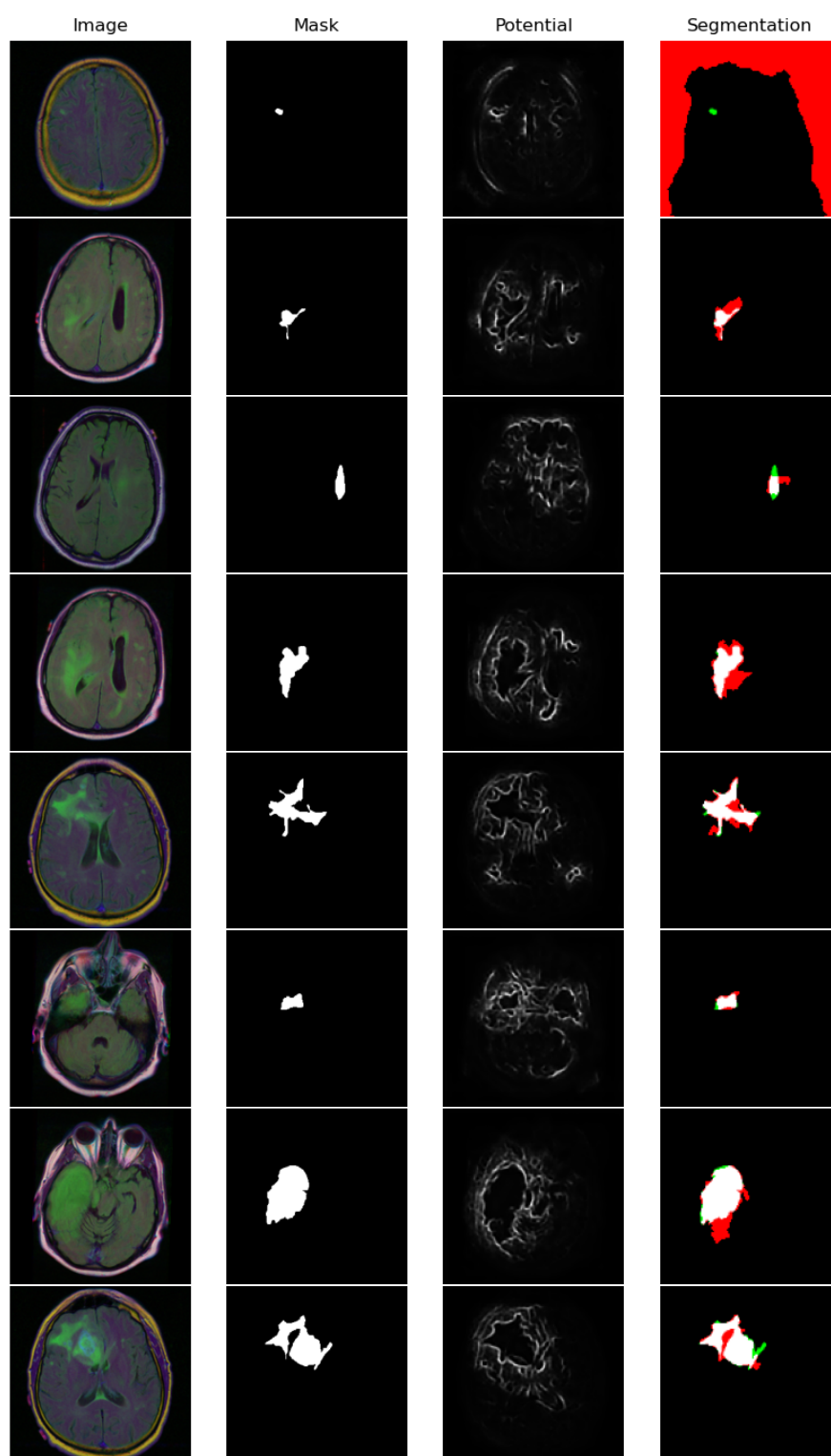


Figure 3.31: Examples where our FMECNN model achieves its lowest scores. From Left to Right, figures show the input image, the target segmentation mask, the potential computed by our trained model, and the proposed segmentation (superposed with the target, red canal is the proposed, green canal is the target, and blue is the intersection).

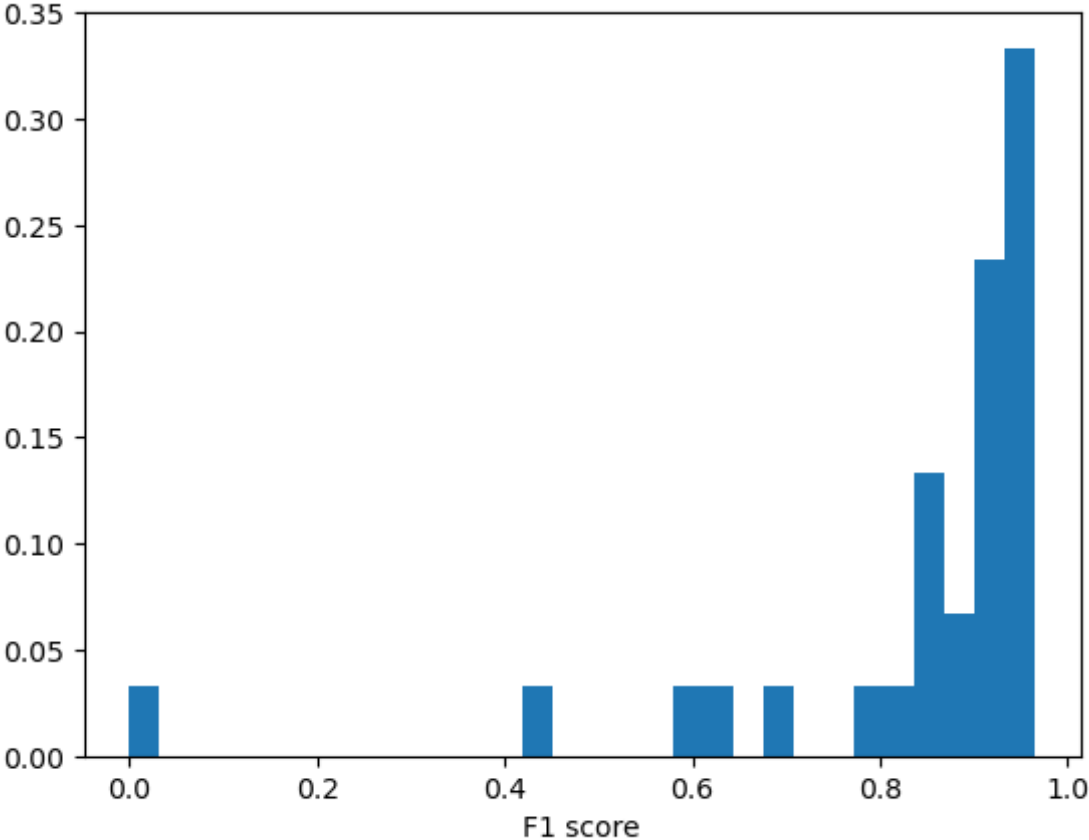


Figure 3.32: Density plot of the F1 score of our model over the test dataset.

3.3.5 Segmenting blood vessels by learning anisotropy

Now we try to tackle the segmentation on another class of images : blood vessels images from the DRIVE dataset. Applying simply the same approach yields results that are not as good as what is expected on such a task. See Figure 3.33 for instance, it shows the kind of output we get, it is easy to see that small vessels aren't properly detected.

Our guess is that including anisotropy in our model might help the model in segmenting those regions. As there is no available algorithm to compute the subgradient in the fashion of the subgradient marching algorithm, we need to use another method for the computation of the geodesic distance and its gradient with respect to the metric tensor. As described in Chapter 2, we can solve the Heat Equation in order to find a kernel explicitly related to the geodesic distance in the small time asymptotic. We used the approximation of the Laplacian operator as the graph Laplacian on the grid like in Heitz et al. 2021.

Also, one should notice the added difficulty to find a point belonging to the vascular network to define as the "seed" and from which to compute the distance map. The regions to segment are very far from being convex and there is no unique way to choose a point in particular. At first, we considered using a database of points of interest like the ones used in our work on landmark detection and geodesic fitting, and it is with those seed points in particular that we achieved the segmentation result shown in Figure 3.33.

Computing the geodesic distance via the heat equation is relatively easy but gets more complicated if we want to compute the distance from a set of points. Indeed our approach can be seen as recovering regions to be segmented $y_{th} \in \{0, 1\}^n$ as $\exp(-d_M(x_0, x_i)^2/4t) \simeq (y_{th})_i$, with x_0 a seed point and x_i a point on the grid. The problem is that even though the region segmented around one of our seed is normalized to sum to 1, it is not clear that the region near a seed will bear the same *mass* around it. Thus to be able to represent y_{th} as the output of our model, it would be required to have an appropriate renormalization constant at hand for every which is not trivial at all, or to compute the geodesic distance by solving the

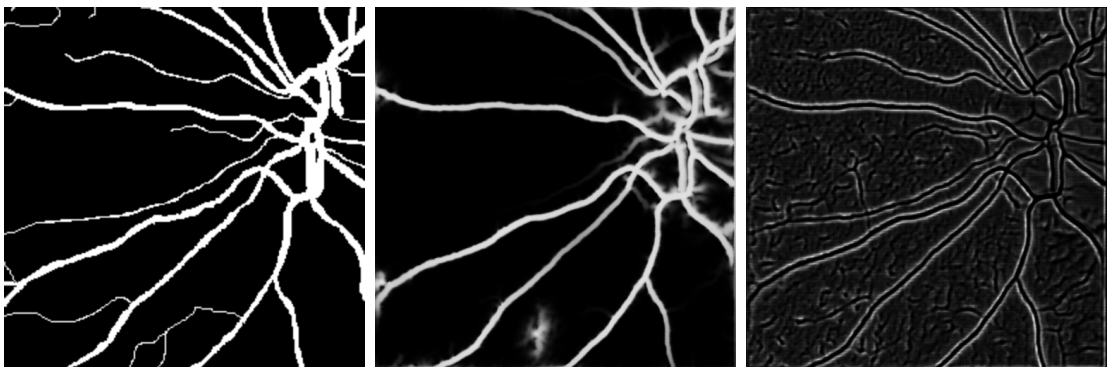


Figure 3.33: First example of generating an isotropic metric with the help of a UNet on an image from the validation set. Left : Ground Truth Segmentation. Center : Proposed Segmentation. Right : Associated Potential output by the UNet.

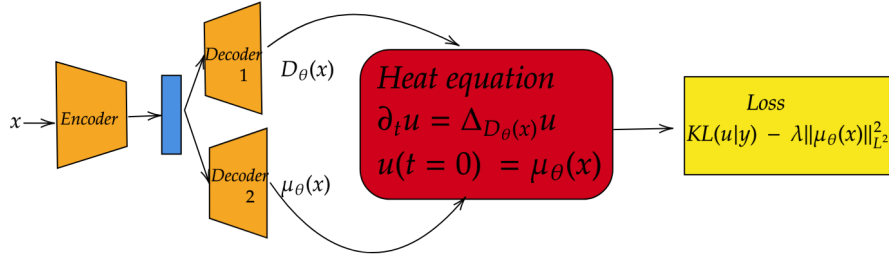


Figure 3.34: Workflow for our segmentation method using Anisotropy.

heat equation for each seed separately and taking the minimum over all the distance maps thus computed, which would be numerically very intensive and long.

Rather, instead of explicitly trying to predict good seeds from the image as the output of the second branch of our UNet, we can simply replace δ_x in the initialisation of the heat flow, instead of simply replacing the position x .

We will thus have a first branch in our network predicting the map $x \mapsto D(x)$ defining the heat flow ($D(x)$ here is the inverse of our classic metric tensor) and a second branch predicting a 2d probability map μ supposedly indicating the location of vascular landmarks or at least points of interest for which the heat flow maps us to an accurate segmentation.

Let (x, y) be couples of input images and associated (normalized) ground truth mask, then the loss we evaluate reads, denoting D_θ the metric tensor output by the first branch of the network and μ_θ the probability map output by the second branch, and also $\Phi_t^D(\nu)$ the heat flow defined by diffusion tensor D applied to $\nu \in \mathcal{P}(\Omega)$ until time t :

$$\mathcal{L}_{seg}(\Phi_t^{D_\theta(x)}(\mu_\theta(x)), y) - \lambda \|\mu_\theta(x)\|_{L^2}^2,$$

with \mathcal{L}_{seg} a chosen loss function for segmentation. As $\Phi_t^D(x)$ is meant to be a probability distribution, we choose a distance function adapted to such object, for instance here we will consider the Kullback-Liebler divergence, other options would include Optimal Transport distances and other such distances that metrize weak-* convergence which is the natural topology for the convergence of (compactly-supported) measures.

We proceed by training our CNN architecture on the DRIVE dataset split into a training and a validation set (60%/40% split) and also test on the IOSTAR dataset, applying random affine transformations and flips on the input image (taking only the green channel) we can avoid overfitting our training set (necessary as there are only 24 images in total in the DRIVE dataset). During our assessment we found that a small batch size of 2 images perform better.

After 250 epochs we reach a DICE score at validation step of 77% which is still lower compared to the state-of-the-art for this task, but already allows to circumvent some of the limitations seen in Figure 3.33.

In Figure 3.35 we show some outputs of the methods. One might notice that

the proposed barycenter is not in fact similar to the barycenter map predicted in the original FMECNN model, but instead it tends to predict a segmentation of the vascular network. This is due to the fact that we don't supervise the second branch of our network to predict seed points directly and the penalizing term in the loss promoting sparsity of the barycenter doesn't collapse the output to a set of a few points (even at larger values of λ). The way we might interpret the computation of the solution to the heat equation from the two outputs of our network, is that it works like an attention mechanism refining the proposed segmentation in the barycenter map.

3.3.6 Partial Conclusion

We have proven that it was possible to generate Riemannian metrics directly from data and to apply geodesic methods in Machine Learning pipelines. We have seen that it is possible to rely on isotropic as well as anisotropic metrics and that the latter present an interest for tasks involving data showing local spatial anisotropy such as vessel segmentation. We hope to open new research questions with these works as it allows the definition of new building blocks for Machine Learning methods. Further works might include the definition of "heat flow" blocks for the definition of ML architectures as we have seen earlier in order to investigate the properties of such an operation as some kind of "geometric" equivalent to the convolution (in the line of what has already been tried with Geodesic Convolutional Networks).

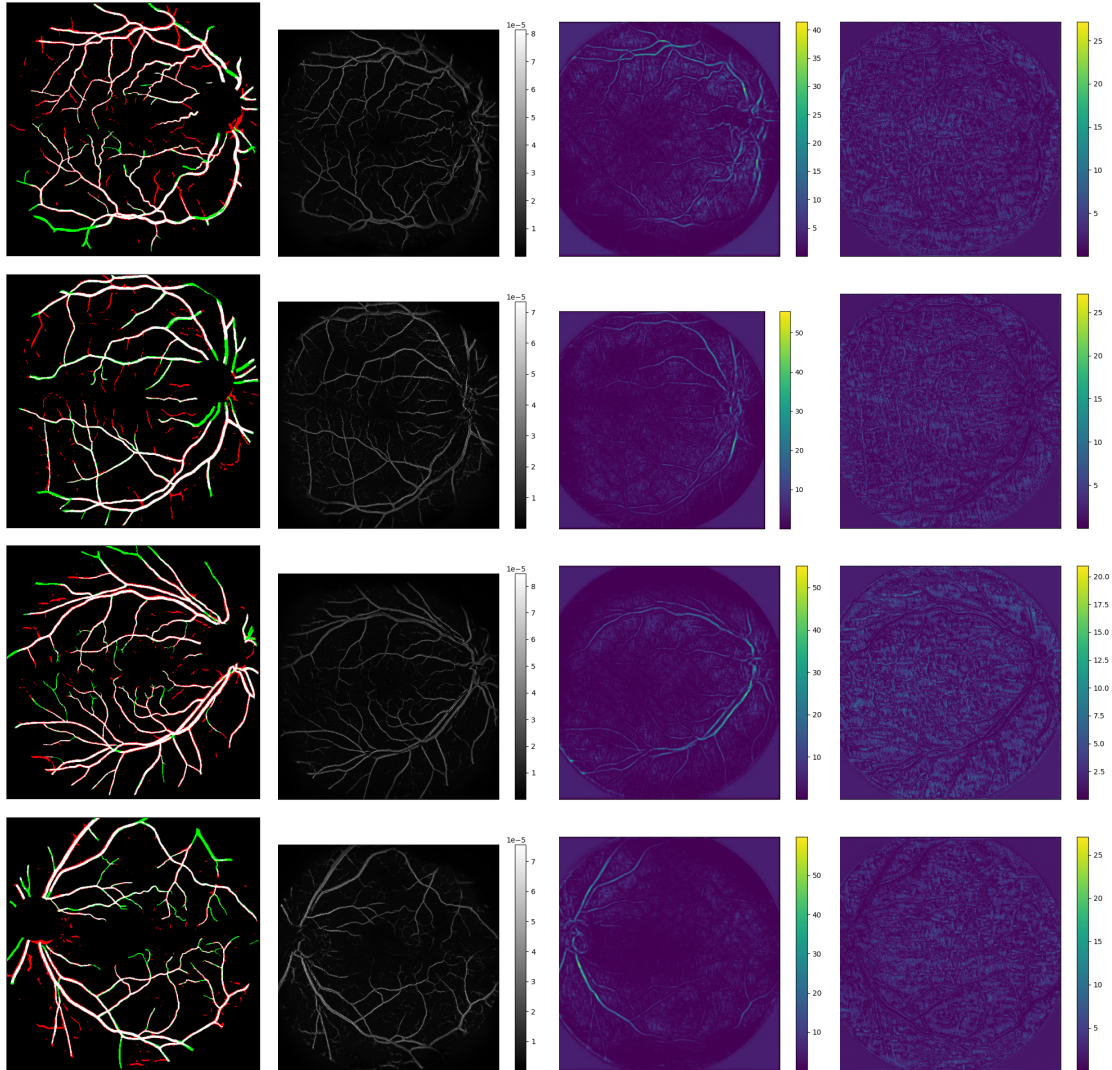


Figure 3.35: Output of our method on a sample from the IOSTAR dataset. Left : Comparison of proposed segmentation versus Ground Truth. Center Left : Barycenter map output by the network. Center Right : Sum of the metric elements in both directions. Right : (log of) Anisotropy factor.

Analysis on the space of paths and Optimal Transport

Optimal Transport and its many faces has seen a huge raise in interest over the last few years.

This chapter is built as a way to naturally lead to our application of Benamou-Brenier regularization in the last sub-sections. Its goal is to quickly introduce the theory of Optimal Transport of mass in order to give a taste of the general context for the definition of the Inverse Problem studied in the second section of this Chapter, by introducing the different point of views on the transport of mass and the relationships between those point of views.

In the following we will consider probability measures on the set Ω , the set of probability measures will be noted $\mathcal{P}(\Omega)$ and the set of Radon measures on Ω will be written $\mathcal{M}(\Omega)$.

If not specified otherwise, one may assume that we will be working with the Borel σ -algebra on Ω .

Measures are usually defined through the axioms of Measure Theory, and Radon measures as a certain level of regularity of such measures. Another point of view that is common among the literature in Analysis is to use the topological dual of evanescent continuous functions on the set Ω . The two definition are related via the Riesz-Markov-Kakutani Representation Theorem.

Given $\mu \in \mathcal{M}(\Omega)$, and a measurable map $T : x \in \Omega \mapsto T(x) \in Y$ we denote by $T_{\#}\mu \in \mathcal{M}(Y)$ the *push-forward measure* of μ by T . Usually its is defined in Measure Theory or Probability textbooks via the inverse image of measurable set as : for any measurable $U \subset Y$, $T_{\#}\mu(U) \stackrel{\text{def.}}{=} \mu(T^{-1}(U))$. Alternatively and in the context of duality between Radon measures and Continuous functions, it may be defined as the adjoint map to right-side composition : $\forall \varphi \in C_0(Y)$, $\langle T_{\#}\mu, \varphi \rangle \stackrel{\text{def.}}{=} \langle \mu, \varphi \circ T \rangle$.

4.1 Introduction to Optimal Transport

Optimal Transport is usually introduced first via the Monge Problem and its generalization into the Monge-Kantorovitch problem. We will quickly overview the

definition of Optimal Transport, its dynamic formulation and its relation to curves, underlying the idea of transport of matter.

4.1.1 The Monge and Kantorovitch problem

First, let us introduce the Monge problem :

Definition 4.1. Let $\mu, \nu \in \mathcal{P}(\Omega)$ be two probability measures on Ω , a measurable map $T : x \in \Omega \mapsto T(x) \in \Omega$ is said to be a solution to the Monge Problem if it is a minimizer for the optimization problem :

$$\inf_{T \# \mu = \nu} \int_{\Omega} c(x, T(x)) \, d\mu(x), \quad (4.1)$$

with c the ground cost. It is called the *Transport map*.

Remark 4.2. *Of course, in Gaspard Monge's time, measure theory and many other concepts were not yet formalized, and the problem was not defined in terms of source and target measure but rather in terms of source and target sets (with uniform density). Formally in our framework it would correspond to solving the former problem with $\mu = \frac{\mathbb{1}_A}{|A|} \mathcal{L}^d$ and $\nu = \frac{\mathbb{1}_B}{|B|} \mathcal{L}^d$ with A and B the source and target sets, and \mathcal{L}^d the classic Lebesgue measure of dimension d .*

Now we introduce the more general Kantorovich problem :

Definition 4.3 (Kantorovich Problem and Optimal Plan). Let $\mu, \nu \in \mathcal{P}(\Omega)$, be two probability measures on Ω , let $\Pi(\mu, \nu) \subset \mathcal{P}(\Omega \times \Omega)$ be the set of couplings between μ and ν , i.e. the measures $\pi \in \Pi(\Omega \times \Omega)$ such that for every measurable set A , $\int_A \int_{\Omega} d\pi(x, y) = \mu(A)$ and $\int_{\Omega} \int_A d\pi(x, y) = \nu(A)$. Then we say that π is an *optimal transport plan* between μ and ν if it is a minimizer for the following optimization problem :

$$\inf_{\pi \in \Pi(\mu, \nu)} \int_{\Omega \times \Omega} c(x, y) d\pi(x, y), \quad (4.2)$$

with c the ground cost.

We will usually ask for c to be continuous, which is sufficient hypothesis to ensure existence of a minimizer : compactness of $\Pi(\mu, \nu)$ and continuity of the objective function are all we need.

We immediately notice that if T is a solution to (4.1), it is also a solution to (4.2) by the transformation $\pi = (id, T) \# \mu$. It is not true in general that there is an Optimal Transport map defining the plan as we will see further when stating Brenier's theorem.

Now let's look at two very easy examples. The first one gives a case where there is no solution to the Monge problem but the solution is obvious in the context of the Kantorovich problem.

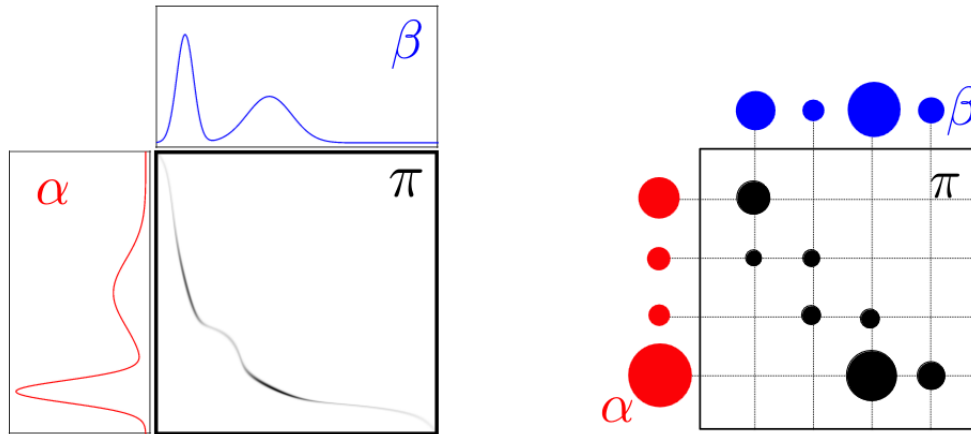


Figure 4.1: Illustration of the OT problem between two probability distributions. Left : the continuous case. α and β are source and target distributions, π is the optimal coupling. Right : the discrete case. Taken from Peyré and Cuturi 2020.

Example 7. Let $\Omega = \mathbb{R}$, let $\mu = \delta_0$ and $\nu = \frac{1}{2}\delta_{-1} + \frac{1}{2}\delta_1$, then there's no optimal transport map in the sense of the Monge Problem, but it is easy to see that the transport plan $\pi = \frac{1}{2}\delta_{(0,-1)} + \frac{1}{2}\delta_{(0,1)}$ is a minimizer for the Kantorovich problem with cost $c = \|\cdot - \cdot\|^2$.

We see that the Monge Problem does not allow to split mass between two locations.

Our second example is simply here to illustrate the non-uniqueness of the solutions to the Kantorovich formulation.

Example 8. Let $\Omega = \mathbb{R}^2$, let $\mu = \frac{1}{2}\delta_{(0,0)} + \frac{1}{2}\delta_{(1,1)}$ and $\nu = \frac{1}{2}\delta_{(1,0)} + \frac{1}{2}\delta_{(0,1)}$ two measures supported symmetrically on the opposite corners of the unit square. We consider the Kantorovich problem with $c(x, y) = \|x - y\|^2$. Then any of the couplings between μ and ν will achieve the minimum. One may also notice that if we restrict ourselves to Monge maps, we get only two different solutions.

The Optimal Transport problem is important because it allows us to define a good distance between measures and is very useful to compare geometric objects represented by measures. It also allows us to define some PDEs as gradient flows in the space of measures equipped with the Wasserstein distance (i.e. the Kantorovich problem with cost $d(x, y)^p$ where d is a distance on the ground space, see Ambrosio, Gigli, et al. 2008 for a textbook on the subject).

To conclude our tour of the definition of "static" Optimal Transport we can state a theorem by Brenier in Brenier 1991 that will give sufficient condition for the existence a Monge map in the case of 2-Wasserstein Optimal Transport (which will be our main interest). It was later extended to more general cases in Ambrosio and

Rigot 2004; Bertrand 2008; Gigli et al. 2012; Ambrosio and Rajala 2014; Cavalletti and Huesmann 2015.

Theorem 4.4 (Brenier 1991). *Let $\Omega = \mathbb{R}^d$ and $c(x, y) = \|x - y\|^2$, let one of the two marginals $\mu = \rho \mathcal{L}^d$ be absolutely continuous with respect to the Lebesgue measure, then there exists a unique optimal transport plan between μ and ν , and it is given as a Monge map T solving (4.1) (i.e. the transport plan is given by $\pi = (\text{id}, T)_\# \mu$).*

Furthermore, this Monge map is unique and given as the gradient of a unique convex function (up to an additive constant).

Now that we have further explained the relationship between the two former optimization problem, we will state two results motivating our interest for the Optimal Transport distances in Machine Learning.

First, for completeness we introduce the weak-* topology

Definition 4.5. Let $(\mu_n) \in \mathcal{M}(\Omega)^{\mathbb{N}}$ be a sequence of Radon Measures, it is said to be convergent in the weak-* topology towards $\mu \in \mathcal{M}(\Omega)$ if and only if $\forall \phi \in \mathcal{C}_0(\Omega), \langle \mu_n, \phi \rangle \rightarrow \langle \mu, \phi \rangle$ where $\mathcal{C}_0(\Omega)$ is the set of continuous evanescent functions defined formally as :

$$\mathcal{C}_0(\Omega) \stackrel{\text{def.}}{=} \left\{ \phi \in \mathcal{C}(\Omega, \mathbb{R}), \quad \forall \epsilon > 0, \exists K_\epsilon \text{ compact s.t. } \|\phi\|_{\infty, \Omega \setminus K_\epsilon} \leq \epsilon \right\}$$

Theorem 4.6. *Let W_2 be the Optimal Transport cost between measure with ground cost $c(x, y) = d(x, y)^2$ for a distance d on the set Ω .*

Then W_2 is a distance on $\mathcal{P}(\Omega)$ and if Ω is a compact set, it metrizes the weak- topology.*

Lemma 4.7. $\delta : x \in (\omega, d) \mapsto \delta_x \in (\mathcal{P}(\Omega), W_2)$ is an isometry.

One can refer to Villani 2021; Villani 2009 for a complete overview of the properties of Optimal Transport.

4.1.2 The Dynamic Formulation

One of the most interesting property of the Optimal Transport problem for applied mathematicians and physicists is its relation to mechanics through the dynamic formulation that we will be introducing next.

Indeed the 2-Wasserstein Optimal Transport problem may be reformulated in terms of the minimization of a velocity field coupled with a transient measure via the continuity equation. This result was first proven in Benamou and Brenier 2000. Here we cite Theorem 5.28 from Santambrogio 2015 which is more general (but restrained to the case of compact and convex domains).

Theorem 4.8 (Santambrogio 2015, Theorem 5.28). *Let Ω be a compact and convex domain, $\mu, \nu \in \mathcal{P}(\Omega)$, we have*

$$W_p^p(\mu, \nu) = \min \{ \mathcal{B}_p(\rho, E), \quad \partial_t \rho + \nabla \cdot E = 0, \quad \rho_0 = \mu, \quad \rho_1 = \nu \}, \quad (4.3)$$

with $(\rho, E) \in \mathcal{C}([0, 1], \mathcal{M}(\Omega) \times \mathcal{M}(\Omega)^d)$ and

$$\mathcal{B}_p(\rho, E) = \frac{1}{p} \int_0^1 \|v_t\|_{L^p(\rho_t)}^p dt$$

if $E = v\rho$ and $+\infty$ elsewhere.

In Equation (4.3), the equation is to be understood in the weak sense.

This formulation resembles classic problems in physics and mechanics of displacement of solids where particles tend to minimize their kinetic energy. The continuity equation appearing in the constraints in (4.3) is ubiquitous in the modeling of the mechanics of matter (be that in Fluid Mechanics or Mechanics of the solid).

4.1.3 Optimal Transport as measures on the space of paths

To show the formulation of interest of the Optimal Transport problem, we will need a few notions from the theory of Length and Geodesic Spaces.

Definition 4.9 (Absolutely Continuous curves). *Absolutely Continuous curves* are curves $\gamma \in C([0, 1], X)$ with values in a Length Space (X, d) for which there exist a function $h \in L^1([0, 1], \mathbb{R})$ such that :

$$\forall 0 \leq s \leq t \leq 1, \quad d(\gamma(t), \gamma(s)) \leq \int_s^t h(u) du. \quad (4.4)$$

The set of such curves is denoted $AC([0, 1], X)$, moreover if one of the functions h satisfying (4.4) is also an element of $L^p([0, 1], \mathbb{R})$ for some $p \in [1, +\infty]$, we may write that $\gamma \in AC^p([0, 1], X)$.

Common absolutely continuous curves include Lipschitz and \mathcal{C}^1 curves. Let us now define the metric analog to the norm of the derivative.

Definition 4.10 (Metric derivative). We may define the *metric derivative* of a curve $\gamma \in AC([0, 1], X)$ by

$$\forall t \in [0, 1], \quad |\dot{\gamma}|(t) = \lim_{\eta \rightarrow 0} \frac{d(\gamma(t), \gamma(t + \eta))}{|\eta|} \in [0, +\infty], \quad (4.5)$$

and it is the optimal function satisfying (4.4) as in definition 4.9.

It is easy to see that for a differentiable curve this definition indeed corresponds to the definition

Definition 4.11 (Length and Geodesic Space). A *Length Space* is a metric space (X, d) such that

$$\forall x, y \in X, \quad d(x, y) = \inf_{\gamma \in AC, \gamma(0)=x, \gamma(1)=y} \int_0^1 |\dot{\gamma}|(t) dt. \quad (4.6)$$

Let $Geo(X)$ be the set of minimizing curves between its two endpoints in the former objective function, and arc-parameterized, thus entirely characterized by

$$\int_0^1 |\dot{\gamma}(t)|^2 dt = d(\gamma(0), \gamma(1))^2. \quad (4.7)$$

A *Geodesic Space* is a Length space (X, d) such that for all $x, y \in X$, there exist $\gamma \in Geo(X)$ with $\gamma(0) = x$ and $\gamma(1) = y$.

Remark. Please note that in metric theory *geodesics* are not solutions of any ODE like in Riemannian Geometry, but simply length-minimizing curves. In Riemannian Geometry on the contrary, *geodesics* are defined as solutions of a particular ODE and are only locally minimizing in general. We will try to be as clear as possible when confusion might arise.

Let X be a geodesic space, then for any $(x, y) \in X^2$, we know there is a geodesic curve with constant time parametrisation γ linking x to y . Then the square of the distance function may be written again in terms of the *Quadratic action* or *Kinetic Energy* of γ :

$$d(x, y)^2 = \int_0^1 |\dot{\gamma}(t)|^2 dt.$$

Integrating against any probability measure π on $X \times X$ we write again the Optimal Transport objective function :

$$\int_X \int_X d(x, y)^2 d\pi(x, y) = \int_X \int_X \int_0^1 |\dot{\Gamma}(x, y)(t)|^2 dt d\pi(x, y),$$

with $\Gamma : (x, y) \mapsto \Gamma(x, y) \in Geo(X)$ a map selecting a geodesic curve between x and y .¹

Then writing $\mathcal{A}_2(\gamma) = \int_0^1 |\dot{\gamma}(t)|^2 dt$ the quadratic action, one can write again the objective function as

$$\int_{\Gamma} \mathcal{A}_2(\gamma) d\Gamma_{\#}\pi,$$

where $\Gamma_{\#}\pi \in \mathcal{M}(\Gamma)$.

The Optimal Transport may thus be rewritten :

$$W_2^2(\mu, \nu) = \min_{\sigma \in \mathcal{P}(\Gamma), (e_0, e_1)_{\#}\sigma = (\mu, \nu)} \int_{\Gamma} \mathcal{A}_2(\gamma) d\sigma(\gamma),$$

This means that, implicitly when trying to find an optimal transport plan from one source measure to another target measure, we are in fact considering every possible continuous path (if the base space allows) from the support of the first measure to the other one.

Let us cite directly Theorem 9.13 from Ambrosio, Brué, et al. 2021 to have a formal statement to the former :

¹Existence of such a geodesic selection map is ensured by Theorem 6.9.13 in Bogachev 2007, and the fact that the multi-valued map $(x, y) \mapsto \gamma_{x,y}$ has a closed graph.

Theorem 4.12 (Ambrosio, Brué, et al. 2021, Theorem 9.13). *If (X, d) is a Polish space and a Geodesic Space, then*

$$\min_{\pi} \int_{X \times X} d(x, y)^2 d\pi(x, y) = \min_{\sigma \in \mathcal{P}(\Gamma)} \int_{\Gamma} \mathcal{A}_2(\gamma) d\sigma(\gamma).$$

Additionally, $\sigma \in \mathcal{P}(\Gamma)$ is optimal if and only if it is supported on $\text{Geo}(X)$ and $(e_0, e_1)_{\#}\sigma \in \Gamma_o(\mu, \nu)$.

The relationships between the three formulations are summarized in 4.2. For a proper definition of the map Θ that relates the Transport problem set in the Length space to the dynamic problem, we refer to Duval and Tovey 2022, Theorem 2.2.

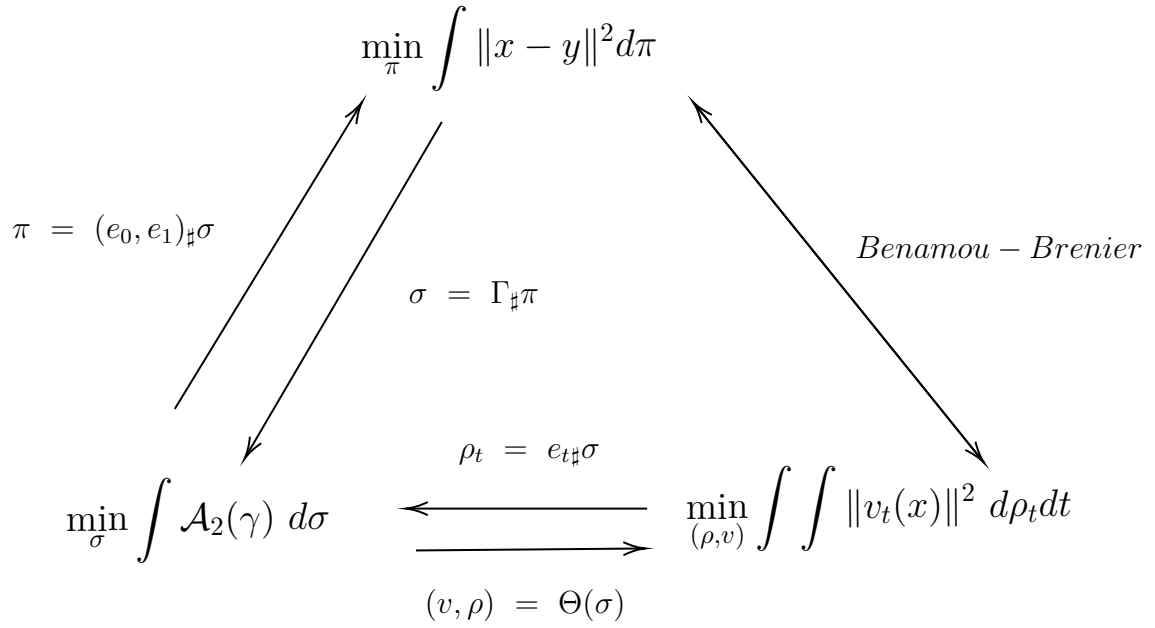


Figure 4.2: Summary of the relationships between problems related to 2-Wasserstein Optimal Transport

4.2 Benamou-Brenier regularization for off-the-grid Curve recovery in the space of positions and orientations

The following section exposes an ongoing work carried out jointly with Bastien Laville, Laure Blanc-Féraud and Gilles Aubert.

4.2.1 Introduction and contributions

This work introduces a functional designed for off-the-grid tracking of point sources. In the context of inverse problems, this involves recovering dynamic Dirac measures. The interest in this area stems from recent advancements in the off-the-grid community, as seen in studies by Bredies, Carioni, Fanzon, et al. 2021; Bredies, Carioni, Fanzon, et al. 2022; Duval and Tovey 2022 and Duval and Tovey 2022, which have introduced methods for recovering moving point trajectories and their numerical implementation.

However, current literature lacks a tractable off-the-grid framework for complex paths where spikes cross each other. Existing state-of-the-art methods fail to accurately reconstruct these scenarios, instead producing separate paths. Such situations naturally occur in various fields, including biomedical imaging, where objects used in super-resolution, like air bubbles, may intersect. An example of these curves in a practical setting is shown in Figure 4.3.

We propose a method to recover point source paths even when crossings occur. Additionally, we present a general theoretical framework for the numerical aspects, ensuring convergence of the discretized problem to the continuous problem under fewer assumptions. To the best of our knowledge, this is the first attempt to recover tangled paths of dynamic Dirac measures in an off-the-grid manner.

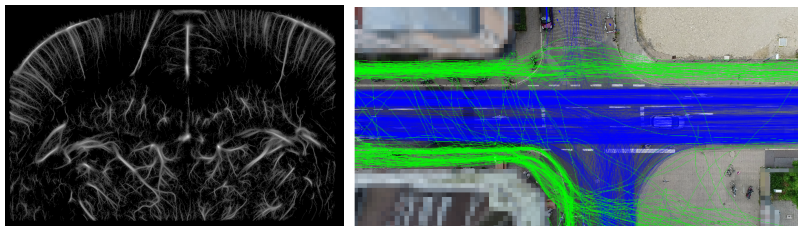


Figure 4.3: Curves arise genuinely in several biological structures or data application, such as cell nucleus, blood vessels, GPS tracks, *etc.*

Figure 4.4 shows an example of an output of the methods in the literature for the simple case of two curves crossing in the middle of the domain. Usually, those failures to properly reconstruct the support of the curves are overlooked, as the projection on the space-time cylinder $\rho_t = e_{t\sharp}\sigma$ is not far from the ground truth projection. But as we have seen in the contributions presented in the manuscript, it might be interesting for some tasks to have access to properly reconstructed supports

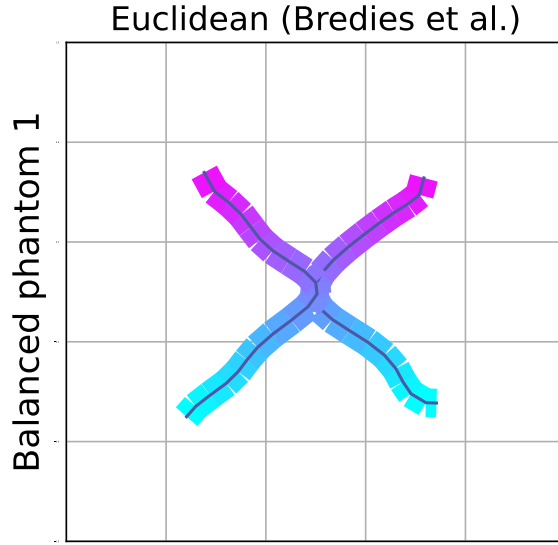


Figure 4.4: Reconstruction of the first phantom of Bredies and Fanzon 2019; Duval and Tovey 2022 consisting of a stack of 21 images showing two crossing curves at constant speed, passed through the acquisition process and then altered by 60% of Gaussian noise. Result from Bredies and Fanzon 2019, as reported in Duval and Tovey 2022, the crossing cannot be recovered.

in order to define (for instance) orientation-dependent features, which would fail in the center of the domain shown in Figure 4.4.

We will consider an inverse problem model similar to the one proposed in Bredies and Fanzon 2019 for which Duval and Tovey 2022 proposed numerical recipes, on the other hand Schmitzer et al. 2019 seems to be the first work to propose the considered regularization. The main idea is to solve an inverse problem by building a measure supported on a space of curves minimising an energy functional with a data term that penalises distance between data and the candidate minimiser, and a regularisation term that is linear in the candidate minimising measure. Such formulation allows the user to yield guarantees over the shape of the minimisers. Indeed, this problem was derived having in mind the opportunity to use a *representer theorem* as the ones proven in Boyer et al. 2019; Bredies and Carioni 2019 as it ensures that the shape of the minimiser is determined by the extreme points of the unit ball of the regulariser. The use of the *representer theorem* requires the fulfilment of a few hypotheses, among which are the finite-dimensionality of the space on which the distance to data is computed. In Bredies and Carioni 2019, the authors give a proof that the extreme points of the unit balls for the considered regularisation are Diracs measures on the space of curves, this ensures the sparsity of the solution. One may note that if Bredies and Carioni 2019 and Bredies, Carioni, Fanzon, et al. 2021 express the problem in terms of a time-dependent density (represented as a measure-valued continuous function of time) and a velocity field, coupled via the continuity equation, and that realise the optimal transport between two marginal distributions. If the correspondence is not one-to-one, one can associate a measure paths to a density and velocity field couple, and conversely, at least under suitable smoothness conditions.

On the numerical side, a few works have been trying to propose numerical methods to recover curves from a temporal series of measurements. For instance, Bredies, Carioni, Fanzon, et al. 2022 proposes a *conditional gradient method* or *Frank-Wolfe algorithm* Frank and Wolfe 1956 for solving the considered inverse problem. *Frank-Wolfe* type algorithms are typical algorithms to solve an optimisation problem under constraints, in an iterative manner and in order to exploit the nature of the considered problem. The idea is to iteratively minimise the linearised version of the objective function on a compact set. Classic theory tells us that such minimizers will be located on extreme points of this compact set. Latest developments Duval and Tovey 2022 extends on the idea of using Frank-Wolfe to solve the inverse problem and leverage a dynamical programming approach to speed up the computation. They also extend the numerics to the 'unbalanced' case where each curve has a time-dependent amplitude associated.

Contributions

Our main contributions in this article are listed below:

- Lift the curve recovery problem in the position-orientation space;
- Introduce a new regularisation term for dynamic Dirac measure recovery, in particular for crossing trajectories;
- Propose an efficient greedy algorithm, with several numerical examples to illustrate the capabilities of our proposed method.
- Prove the Γ -convergence of the discretised curve problem to the global one, thus showing the interest of discretising the space of curves using Bézier curves as approximations in the practical context;

Outline

This section is organised as follows : the first subsection introduces the subject with as little mathematics and jargon as possible and gives an idea of what has been done and what we contribute to the literature, the second subsection introduces the variational problem, formulation and first numerical results on synthetic data, then subsection 3 gives theoretical insights as to why the proposed discretisations work. Finally, the last subsection show our attempts to bring our method to real world data in biomedical task.

4.2.2 An energy for trajectory recovery

An *inverse problem* aims to recover some physical quantities, from a low-pass filtered observation. This is typically the localisation of sources, from a blurred or downgraded image. The *off-the-grid* variational methods, also called *gridless* methods, are a rather recent addition to the literature Bredies and Pikkarainen 2012; Castro and Gamboa 2012; Duval and Peyré 2014; Denoyelle et al. 2016, designed to overcome the limitations of the discrete methods and more precisely by the fine

grid. Indeed, in the discrete framework such as the Basis Pursuit/LASSO, a fine grid is introduced and the sources are estimated within this setting: the point (also called *spikes*) are thereby constrained on this grid, further yielding discretisation discrepancies. On the contrary, off-the-grid methods do not rely on such grid and rather consist in the optimisation of a measure in terms of amplitude, position and number of the point sources. Moreover, they offer several theoretical results, such as a quantitative bound characterising the discrepancies between the source and the reconstruction.

Let's describe the Inverse Problem setup for the trajectory recovery of contrast agents. We suppose we have access to noisy measurements $b_t \in \mathcal{H}$, indexed by time $t \in [0, 1]$ and lying in a finite-dimensional Hilbert space \mathcal{H} and that they were obtained through an acquisition process defined by a linear operator Φ that maps $\mathcal{M}(\Omega)$ to \mathcal{H} .

Let $T \in \mathbb{N}^*$ be the number of time samples, further defining $(t_i)_{1 \leq i \leq T}$ the time slices ranging from 0 to 1. The following functional is the core component of the dynamic off-the-grid framework Bredies, Carioni, Fanzon, et al. 2021; Bredies and Fanzon 2019 so far:

$$\operatorname{argmin}_{\sigma \in \mathcal{M}(\Gamma)} E(\sigma) = \operatorname{argmin}_{\sigma \in \mathcal{M}(\Gamma)} \sum_{i=1}^T \|A_i e_{t_i \#} \sigma - b_{t_i}\|_{\mathcal{H}}^2 + \int_{\Gamma} w(\gamma) d\sigma(\gamma), \quad (4.8)$$

- $\Gamma \subset \left\{ \gamma = (h, \xi), \quad h \in C([0, 1], \mathbb{R}), \quad \xi : [0, 1] \rightarrow \Omega, \quad \xi|_{h \neq 0} \text{ is continuous} \right\}$ the space of curves considered,
- With e_t the measurable map of evaluation at time t , $e_t(\gamma) = \gamma(t)$ i.e. $e_{t \#} \sigma \in \mathcal{M}(\Omega)$,
- $A_i : \mathcal{M}(\Omega) \rightarrow H$ a linear operator, pertaining to the physical context,
- $w : \Gamma \rightarrow \mathbb{R}_+$, a lower semicontinuous map, in this work we will only consider $w(\gamma) = \int_0^T \alpha + \beta |\dot{\gamma}|^2(t) dt$ for $\alpha, \beta > 0$ where $|\dot{\gamma}|$ denotes the metric derivative of γ .
- $t \mapsto b_t \in \mathcal{H}$ are the acquired data called the *observation*.

The first term is a data term, controlling the distance from the acquisition b_t in the acquisition space \mathcal{H} . The second term is the regularization term enforcing the structure upon the minimizer by mean of the Representer Theorem and selecting particular solutions of the inverse problems, favoring "regular" solutions.

In this work we will not depart from the setting used in the literature that uses $\mathcal{H} = \mathbb{R}^n$, and $\forall i \in \llbracket 1, T \rrbracket$, $A_i : \rho \in \mathcal{M}(\Omega) \mapsto \left(\int_{\Omega} a_i^j(x) d\rho(x) \right)_{1 \leq j \leq n} \in \mathcal{H}$. Such finite dimensional space for the data term is of course motivated by the use of the Representer Theorem (Theorem 4.13) as it ensures that the final minimizer will be composed of a finite set of extreme points.

In the following we will be only focusing on the case where Γ is a subspace of $\{\gamma = (1, \xi), \quad \xi : [0, 1] \rightarrow \Omega \in \mathcal{C}^0\}$ (which is a complete space for the uniform convergence), curves with constant amplitude equal to 1, which corresponds to the

Benamou-Brenier case in Duval and Tovey 2022. Although, for the specific case of $w(\gamma) = \alpha + \int_0^1 |\dot{\gamma}|(t)dt$ (either in the Euclidean or Riemannian setting), one can restrict Γ to the set of Absolutely Continuous curves, adding the condition $\xi \in AC^2([0, 1])$, else $w(\gamma) = +\infty$.

As described in the introduction of this Chapter, the problem (4.8) is related to the time-space cylinder definition of the problem through the relationships between the different formulations of the Optimal Transport problem as illustrated in Figure 4.2.

A general Representer Theorem

Representer Theorems are common results in Inverse Problems, usually allowing to derive some structure of the solutions to the optimization problem at hand. Quite recently the authors of Boyer et al. 2019 have been able to establish a very general statement for a Representer Theorem that allows to deal with many different problems.

Theorem 4.13 (Representer Theorem, Bredies and Carioni 2019). *Assuming that the space E is locally convex, $f : \mathcal{H} \rightarrow \mathbb{R}$ and $R : E \rightarrow [0, +\infty]$ are convex, that $\Phi : E \rightarrow \mathcal{H}$ verifies $\Phi(\text{dom}R) = \mathcal{H}$ and R is coercive on the Quotient Space $E/\{R(u) = 0\}$,*

$$\min_{u \in E} f(\Phi u) + R(u),$$

has minimizers decomposable as a finite convex combinations of extreme points of $\{u \in E, R(u) \leq 1\}$ (up to scaling if necessary).

This result is very important as it relates the properties - and in particular geometries - of the minimizers to the regularizer R , more specifically to the extreme points of its balls (or sublevel sets). Immensely popular examples include the L^1 regularization of the problem that promotes sparsity of the solution (associated with dirac measures as extreme points) and TV regularization ($TV : u \mapsto \sup_{\phi \leq 1} \langle \text{div} \phi, u \rangle$ yields characteristic functions of regular sets, up to normalization). More recently the authors of Laville et al. 2021 have proposed a regularizer that also selects curves, this time in the static framework.

As explained before, Theorem 4.13 is to be understood as a general result among a diverse family of Representer Theorems that allow to find such sparse decompositions. Another very general formulation, and important to mention, is the one presented in Boyer et al. 2019, with slightly different hypotheses.

4.2.3 The orientation-position space

As we mentioned earlier, the original curve reconstruction model struggles to recover multiple crossing curves in 2D. Indeed, the regularising term in (4.8) selects solutions with minimal length in the 2D plane, thus preferring couples of nearly-touching curves to crossing curves. To prevent such issue, we propose an approach consisting in lifting our problem in the space of position and orientation, adding an orientation coordinate to our 2D variable.

Hence, we lift the problem in the so-called roto-translation space, inspired by the variational approach proposed by Chambolle and Pock 2019. The orientation-position space, also called the roto-translation space in the literature, is understood as homogeneous space upon which the Special Euclidean group $SE(d)$ acts transitively and faithfully, and further identified here as $\mathbb{M}_d \stackrel{\text{def.}}{=} \mathcal{X} \times \mathbb{S}^{d-1}$. As mentioned earlier, $\mathcal{X} \subset \mathbb{R}^d$ is then the spatial domain where the acquisition is defined, and \mathbb{S}^{d-1} thereby parametrises the local orientation of the curve. Let us introduce $SO(d)$ the d -dimensional rotation group, *i.e.* the group of all rotations about the origin of d -dimensional Euclidean space \mathbb{R}^d under the operation of composition. Then, we introduce:

Definition 4.14. The homogeneous space of position-orientations in \mathbb{R}^d is denoted \mathbb{M}_d :

$$\mathbb{M}_d \stackrel{\text{def.}}{=} \mathbb{R}^d \rtimes SO(d)/\{0\} \rtimes SO(d-1).$$

It is $(2d - 1)$ -dimensional and exhibits the nice property that $SE(d)$ acts transitively and faithfully on it. In the following we use $d = 2$, given our problem is set on 2-dimensional images. This space is useful because it allows one to differentiate objects with the same position but with different orientations, which is a common feature in $2D$ vessel images, as the projection in a $2D$ image of an inherently tree-like structure gives rise to crossings.

If needed, we will identify $n = (\cos(\theta), \sin(\theta)) \in \mathbb{S}^1 \longleftrightarrow \theta \in \mathbb{R}/\mathbb{Z} \longleftrightarrow R_\theta \in SO(2)$. As of now, $N \in \mathbb{N}^*$ control points will be no longer on the \mathcal{X}^N but rather in the $(\mathbb{M}_d)^N$ to further incorporate the orientation information in the curve. The roto-translational curve space is now:

$$\Upsilon \stackrel{\text{def.}}{=} \left\{ \gamma = (h, \xi), \quad h \in C([0, 1], \mathbb{R}), \quad \xi : [0, 1] \rightarrow \mathbb{M}_d, \quad \xi_{|h \neq 0} \text{ is continuous} \right\}$$

and the measure space $\mathcal{M}(\Upsilon)$ follows naturally. Eventually, note that the original roto-translational space \mathbb{M}_d can also be seen as a Riemannian manifold equipped with a metric tensor, which ought to be defined in the following to further implement the regularisation on curves crossing.

In this manuscript we only treat the "Benamou-Brenier" case where the mass or amplitude transported along the curve, denoted by h , is constant and equal to 1. We will only work in the subset of Υ of curves (h, ξ) with $\forall t \in [0, 1], h(t) = 1$.

A relaxed Reeds-Shepp metric

Multiple models have taken advantage of the space of roto-translation \mathbb{M}_d , among which applications we could quote computing geodesics taking into account orientation features and penalising curvature. Usually used to model the movements of vehicles or oriented objects in \mathbb{R}^d , the Reeds-Shepp metric Reeds and Shepp 1990 has numerous applications in various context. It is a sub-Riemannian metric that prevents curves to not be planar and penalises changes of direction (meaning penalising curvature of the trajectory). We will use a relaxed Riemannian version \mathcal{F}_ε

of the Reeds-Shepp metric \mathcal{F} here that only penalises velocities orthogonal to. This model continuously converges to the standard sub-Riemannian Reeds-Shepp metric Duits, Meesters, et al. 2018 as the relaxation parameter ε yields 0.

We may cite applications of the roto-translational space in medical images, such as in E. Bekkers et al. 2014 which applies filters on lifted images to help track blood vessels in vascular imaging and in E. J. Bekkers, Duits, et al. 2015 for the computation of geodesics in $SE(2)$. The following definition of \mathcal{F}_ε is peculiar to the biomedical application, formulated with some cost functions introducing another scaling parameter.

Definition 4.15 (Relaxed Reeds-Shepp metric). The relaxed Reeds-Shepp metric tensor field Duits, Meesters, et al. 2018 can be written as a Euclidean norm at each point in \mathbb{M}_2 . Let $0 < \varepsilon \leq 1$ be the relaxation parameter, $\xi > 0$ a scaling parameter, consider $(x, \theta) \in \mathbb{M}_2$ while $(\dot{x}, \dot{\theta}) \in T(\mathbb{M}_2)$ lies in the tangent bundle, then:

$$\begin{aligned} \left\| (\dot{x}, \dot{\theta}) \right\|_g^2 &\stackrel{\text{def.}}{=} |\dot{x} \cdot e_\theta|^2 + \frac{1}{\varepsilon^2} |\dot{x} \wedge e_\theta|^2 + \xi^2 |\dot{\theta}|^2 \\ &= \left\langle R_\theta(\dot{x}, \dot{\theta}), \text{diag} \left(1, \frac{1}{\varepsilon^2}, \xi^2 \right) R_\theta(\dot{x}, \dot{\theta}) \right\rangle, \end{aligned}$$

where e_θ is the unit vector in the direction defined by the angle θ . The idea of this relaxation boils down to, the increasing penalisation of non-planarity of the lifted path as ε nears 0.

The problem further writes down to the optimisation of the following energy:

$$\underset{\sigma \in \mathcal{M}(\Upsilon)}{\text{argmin}} E(\sigma) = \underset{\sigma \in \mathcal{M}(\Upsilon)}{\text{argmin}} \sum_{i=1}^T \|A_i e_{t_i} \# \sigma - b_{t_i}\|_{\mathcal{X}} + \beta \int_{\Gamma} \int_0^1 \|\gamma'(t)\|_g dt d\sigma(\gamma). \quad (4.9)$$

The relaxed version of Reeds-Shepp accurately approaches the sub-Riemannian case with infinite cost for non-planar curves. The $|\dot{\theta}|^2$ term penalises steering the prescribed orientation of the curve, and acts as a penalisation of the local curvature of the curve.

An Unravelling Frank-Wolfe algorithm

In the continuity of the literature on our model of curve recovery, we will use the Frank-Wolfe algorithm to find the minimizer. It is a numerical algorithm proposed 1956 by Marguerite Frank and Philip Wolfe Frank and Wolfe 1956.

The Frank-Wolfe algorithm, also known as the conditional gradient method, is an iterative optimization technique designed for solving constrained convex optimization problems. It operates by linearizing the objective function at the current point and then solving a linear subproblem to find a search direction within the feasible region. This linear subproblem is typically easier to solve than the original nonlinear problem. The algorithm then moves towards the solution of the linear subproblem, updating the current solution iteratively. The key principles of the Frank-Wolfe algorithm are its simplicity in handling constraints and its ability to

exploit the geometry of the feasible region, making it particularly useful for large-scale problems where traditional methods might be computationally infeasible.

Algorithm 4 shows pseudo-code for the Frank-Wolfe algorithm. In the following we will be mostly interested in looking at the curves we are able to find in the Oracle step (line 4 in the algorithm) and we will not use any stopping criterion other than simply after accepting a set number of curves, neither will we focus on determining the amplitude of the Dirac measures to be added to the solutions and we will make the hypothesis that they all have the same weight in the solution.

The Frank-Wolfe algorithm is particularly adapted to optimization over spaces of measures and the use of the Representer Theorem. In our case the structure of the solution of this *Linear Optimisation Oracle* is again directly related to the extreme points of the regularizer (Duval and Tovey 2022, Lemma 4.2, to ensure those nice properties we use $D = \{\gamma/w(\gamma) \leq \lambda\}$ a sub-level set of w with sufficiently large constant λ) and we need to find a solution to the linearized problem as a dirac distribution on a single curve δ_γ . Thus the LMO turns into a non-convex optimization problem of the curve γ , that we propose to approximate by parametrizing the space of curves.

Algorithm 4 : Frank-Wolfe algorithm for curve recovery

Data : Acquisition $t \mapsto b_t \in \mathcal{H}$, number of iterations K , regularisation weight $\beta > 0$.

1 Initialisation: $m^{[0]} = 0$.

2 **for** k , $0 \leq k \leq K$ **do**

3 For $m^{[k]} = \sum_{i=1}^k a_i^{[k]} \delta_{\gamma_i}$ such that $a_i^{[k]} \in \mathbb{R}$,

$$\eta^{[k]}(x) \stackrel{\text{def.}}{=} \frac{1}{\alpha} \Phi^*(\Phi m^{[k]} - y).$$

4 Find $\gamma^* \in \Gamma$ such that :

$$\gamma^* \in \underset{\gamma \in D}{\operatorname{argmin}} \langle \eta^{[k]}, \delta_\gamma \rangle + w(\gamma).$$

5 Find corresponding weight $a_{i+1}^{[k]}$

6 $m^{[k+1]} = m^{[k]} + a_{k+1}^{[k]} \delta_{\gamma_{k+1}}$

7 Compute the amplitudes $\mathbf{a}^{[k+1]} \leftarrow \operatorname{argmin} E(\sum_i a_i \delta_{\gamma_i})$

8 Prune the amplitudes

9 Check STOPPING CRITERION

10 **end**

Result : Discrete measure $m^{[k]}$ where k is the stopping iteration.

4.2.4 Discretisation of the space of curves, beyond polygonal curves

Focusing only on the case of a finite number of timestamps in the data term, and as pointed out in Duval and Tovey 2022, Lemma 3.3, one can always define the following map

$$g : \gamma \mapsto \tilde{\gamma} \in \operatorname{argmin}\{w(h), h \in \Gamma, \forall i, h(t_i) = \gamma(t_i)\}, \quad (4.10)$$

which leads to the inequality

$$E(g_{\#}\sigma) \leq E(\sigma). \quad (4.11)$$

This means that there is a theoretically optimal discretization of the space, as any proposed solution to the optimization problem can be replaced by a better one, only conserving the location of the curve at the timestamps where error in the data term is computed. Namely the optimal curves are simply the curves that are piecewise "geodesic" with respect to w between each of the timestamps t_i .

We will now consider how to discretize the space of curves in order to find a good approximation of the solution of the problem (4.8). In this section we provide a formulation for the approximation of absolutely continuous curves by families of curves defined by finite-dimensional control points.

Let P^n be a map from the space of control points $\mathbb{R}^{d \times k_n}$, and $P_n = P^n(\mathbb{R}^{d \times k_n})$ the n -th order discretisation of the space of curves. Let $\mu \in \mathcal{M}(\mathbb{R}^{d \times k_n})$, then we have that $P_{\#}^n \mu \in \mathcal{M}(\Gamma)$.

Here what we need to ensure in order to guarantee that our notion of discretization is suitable to approximate the theoretical minimizer of the variational problem is to ensure that the problem set in the discretization space effectively converges towards the original problem. The right notion of convergence is Γ -convergence, as defined in the following.

Definition 4.16 (Γ -convergence). Let $F_n : x \in E \mapsto \mathbb{R} \cup \{+\infty\}$ be a sequence of functionals on topological space E .

We say that the sequence F_n converges towards F in the sense of Γ -convergence if and only if :

- $\forall x \in E, (x_n)_{n \in \mathbb{N}} \in E^{\mathbb{N}}, x_n \longrightarrow x \implies F(x) \leq \liminf_n F_n(x_n)$.
- $\forall x \in E, \exists (x_n)_{n \in \mathbb{N}} \in E^{\mathbb{N}}, \text{ s.t. } x_n \longrightarrow x \text{ and } \limsup_n F_n(x_n) \leq F(x)$.

This is a very weak notion of convergence that guarantees that, given a sequence minimizers, if it converges its limit will be a minimizer of the target functional. It is very common in Calculus of Variations and allow to approximate hard or for instance non-convex problems by ones that are much easier to solve, and its applications are both theoretical and numerical.

The following result ensures that, with a few assumptions, the sequence of approximation spaces P_n are good approximations to find the minimizers of (4.8).

Theorem 1. Let $P^n : c \in \mathbb{R}^{d \times k_n} \mapsto P^n(c) \in \Gamma$, $S_n : \gamma \in \Gamma \mapsto S_n(\gamma) \in \mathbb{R}^{d \times k_n}$ be measurable maps, $P^n(\mathbb{R}^{d \times k_n}) = P_n \subset \Gamma$ be an increasing sequence of subsets of Γ , such that $\bigcup_n P_n$ is dense in Γ . Suppose that for every $\gamma \in \Gamma$ $P^n(S_n(\gamma)) \rightarrow \gamma$ for the uniform convergence, and $w(P^n(S_n(\gamma))) \leq w(\gamma)$. Then the functional E defined earlier constrained to P_n Γ -converges to E , formally : $\Gamma\text{-lim}_n(E + \chi_{P_n}) = E$.

Proof. Let $\sigma_n \xrightarrow{*} \sigma$, then as we already have that

$$E(\sigma_n) \leq E_n(\sigma_n),$$

by lower semicontinuity we conclude that

$$E(\sigma) \leq \liminf_n E_n(\sigma_n).$$

For the lim sup inequality, choosing $\sigma_n = (P^n \circ S_n)_\# \sigma \xrightarrow{*} \sigma$, we have that

$$\int w(\gamma) d\sigma_n \leq \int w(\gamma) d\sigma,$$

and thus

$$\limsup_n \int w(\gamma) d\sigma_n \leq \int w(\gamma) d\sigma,$$

and by continuity of F we then get that :

$$\limsup_n E_n(\sigma_n) \leq E(\sigma).$$

□

It is also straightforward to see that we can even lower our hypothesis to simply having $\limsup_n w(P^n \circ S_n(\gamma)) \leq w(\gamma)$, but for the scope of this chapter the hypothesis presented here is sufficient.

Remark. We may notice that the former proof defines an approximation map on P_n . Let $S_n : \gamma \mapsto S_n(\gamma)$ be a measurable map, then $\gamma^n = P^n(S_n(\gamma))$, and we retrieve $\sigma_n = (P^n \circ S_n)_\# \sigma$. If we have $S_n \circ P^n = Id$ as in the case of polygonal lines, it is even a projection.

Remark. The latter result requires no more than very general hypotheses and can be extended to more general settings. For instance, we may consider the continuous time setting where the data term is changed to $\int_0^1 \|Ae_{t\#} \sigma - b_t\|_{\mathcal{H}}^2 dt$, although we lose the application of the Representer theorem that guarantees the shape of the minimizer. Moreover, the only hypothesis on the integrand $w(\gamma)$ is its lower semicontinuity, so we may consider many different regularisers as well.

The former convergence result is to be understood as a kind of criterium in order to answer to the question "what are the minimal assumptions for a discretization of our problem?". Another way to interpret the condition $w(P^n \circ S_n(\gamma)) \leq w(\gamma)$ or $\limsup_n w(P^n \circ S_n(\gamma)) \leq w(\gamma)$ is as a relaxation of (4.11). In a way the inequality is the same, but the important part is that it holds on the "difficult" lower semicontinuous part (the data term being continuous, it does not pose any problem).

Lemma 4.17. *The following operators verify the property $w(P^n(S_n(\gamma))) \leq w(\gamma)$ with $w(\gamma) = \alpha + \beta \int_0^1 |\dot{\gamma}|^2(t)dt$:*

- *Polygonal lines*

$$\forall c \in \mathbb{R}^{d \times k_n}, \quad P^n(c)(t) = \frac{\tilde{t}_{k+1} - t}{\tilde{t}_{k+1} - \tilde{t}_k} c_k + \frac{t - \tilde{t}_k}{\tilde{t}_{k+1} - \tilde{t}_k} c_{k+1}, \quad \text{if } t \in [\tilde{t}_k, \tilde{t}_{k+1}[$$

where the k_n are such that $k_n | k_{n+1}$, that way we ensure that $P_n \subset P_{n+1}$, and $\tilde{t}_k = \frac{k}{k_n}$. We use also $S_n(\gamma)_k = \gamma(k/k_n)$ for $k \in \llbracket 0, k_n \rrbracket$,

- *Bézier curves* $P^n(c)(t) = \sum_{k=0}^n t^k (1-t)^{n-k} \binom{n}{k} c_k$, with $k_n = n+1$ and $S_n(\gamma)_k = \gamma(k/n)$ for $k \in \llbracket 0, n+1 \rrbracket$,
- *Geodesic by parts* $\forall c \in \mathcal{M}^{k_n}$, $P^n(c)(t) = \text{Exp}_{c_k}(k_n(t - \frac{k}{k_n}) \text{Log}_{\mathbb{S}_{c_k}}(c_{k+1}))$, if $t \in [\frac{k}{k_n}, \frac{k+1}{k_n}]$ and $S_n(\gamma)_k = \gamma(k/k_n)$ for $k \in \llbracket 0, k_n \rrbracket$, with $k_n | k_n + 1$ to ensure monotonicity.

Proof. • *Polygonal curves case :*

Taking $(S_n(\gamma))_k = \gamma(\tilde{t}_k)$,

$$\begin{aligned} & \int_0^1 \|P^n(S_n(\gamma))'(t)\|^2 dt \\ &= \int_0^1 \left\| \sum_{k=0}^{k_n-1} \mathbb{1}_{[\tilde{t}_k, \tilde{t}_{k+1}[}(t) (\gamma(\tilde{t}_{k+1}) - \gamma(\tilde{t}_k)) \right\|^2 / (\tilde{t}_{k+1} - \tilde{t}_k)^2 dt \\ &= \sum_{k=0}^{k_n-1} (\tilde{t}_{k+1} - \tilde{t}_k) \left\| \int_{\tilde{t}_k}^{\tilde{t}_{k+1}} \gamma'(s) ds \right\|^2 / (\tilde{t}_{k+1} - \tilde{t}_k)^2 \\ &\leq \sum_{k=0}^{k_n-1} \int_{\tilde{t}_k}^{\tilde{t}_{k+1}} \|\gamma'(s)\|^2 ds \\ &= \int_0^1 \|\gamma'(s)\|^2 ds. \end{aligned}$$

- *Bézier curves :*

4.2. Benamou-Brenier regularization for off-the-grid Curve recovery in the space of positions and on

Taking $(S_n(\gamma)) \in \mathbb{R}^{d \times (n+1)}$, $(S_n(\gamma))_k = \gamma(k/n)$,

$$\begin{aligned}
& \int_0^1 \|P^n(S_n(\gamma))'(t)\|^2 dt \\
&= \int_0^1 \left\| \sum_0^{n-1} \binom{n-1}{k} t^k (1-t)^{n-1-k} \left(\frac{\gamma((k+1)/n) - \gamma(k/n)}{1/n} \right) \right\|^2 dt \\
&\leq \underset{Jensen}{\sum_{k=0}^{n-1}} \binom{n-1}{k} \int_0^1 t^k (1-t)^{n-1-k} \left\| \frac{\gamma((k+1)/n) - \gamma(k/n)}{1/n} \right\|^2 dt \\
&= \sum_{k=0}^{n-1} \binom{n-1}{k} \int_0^1 t^k (1-t)^{n-1-k} \left\| \frac{\int_{k/n}^{(k+1)/n} \gamma'(s) ds}{1/n} \right\|^2 dt \\
&\leq \underset{Jensen}{\sum_{k=0}^{n-1}} \binom{n-1}{k} \int_0^1 t^k (1-t)^{n-1-k} \left(\frac{\int_{k/n}^{(k+1)/n} \|\gamma'(s)\|^2 ds}{1/n} \right) dt \\
&= \sum_{k=0}^{n-1} (1/n) \frac{\int_{k/n}^{(k+1)/n} \|\gamma'(s)\|^2 ds}{1/n} = w(\gamma).
\end{aligned}$$

where we used the properties of the Bernstein polynomial basis : $\sum_k \binom{n}{k} t^k (1-t)^{n-k}$ and $\int_0^1 \binom{n}{k} t^k (1-t)^{n-k} = \frac{1}{n+1}$.

- Piecewise geodesic curves :

with $\tilde{t}_k = k/k_n$

$$\begin{aligned}
& \int_0^1 \|P^n(S_n(\gamma))'(t)\|^2 dt \\
&= \int_0^1 \left\| \sum_{k=0}^{k_n-1} \mathbb{1}_{[\tilde{t}_k, \tilde{t}_{k+1}[}(t) \text{Log}_{\gamma(\tilde{t}_k)}(\gamma(\tilde{t}_{k+1})) \right\|_g^2 / (\tilde{t}_{k+1} - \tilde{t}_k)^2 dt \\
&= \sum_{k=0}^{k_n-1} (\tilde{t}_{k+1} - \tilde{t}_k) \left\| \text{Log}_{\gamma(\tilde{t}_k)}(\gamma(\tilde{t}_{k+1})) \right\|_g^2 / (\tilde{t}_{k+1} - \tilde{t}_k)^2 \\
&= \sum_{k=0}^{k_n-1} d_g(\gamma(\tilde{t}_k), \gamma(\tilde{t}_{k+1}))^2 / (\tilde{t}_{k+1} - \tilde{t}_k) \\
&\leq \sum_{k=0}^{k_n-1} \left(\int_{\tilde{t}_k}^{\tilde{t}_{k+1}} \|\gamma'(s)\|_g ds \right)^2 / (\tilde{t}_{k+1} - \tilde{t}_k) \\
&\leq \underset{Jensen}{\sum_{k=0}^{k_n-1}} \int_{\tilde{t}_k}^{\tilde{t}_{k+1}} \|\gamma'(s)\|_g^2 ds \\
&= \int_0^1 \|\gamma'(s)\|_g^2 ds.
\end{aligned}$$

□

Remark. One may note that the former result is related to Lemma 3.3 and 3.4 in

Duval and Tovey 2022, that states that the minimizer σ^* should be supported on the minimizers of w interpolating points prescribed at the timestamps appearing in (4.8). The corresponding discretisations are the *polygonal lines* and more generally the *piecewise geodesic curves* in the case where $w(\gamma) = \alpha + \beta \int_0^1 |\dot{\gamma}|^2(t)dt$. However, it is still worth looking for other discretisations as it may be numerically helpful during the optimization.

4.2.5 Numerical Results

In this section we will be presenting numerical results associated with the different ideas introduced above. In the interest of transparency, a wider study of the influence of hyperparameters has been carried out as systematically as possible. Section 4.2.10 contains table with the results from experiments carried out with varying parameters of noise, size of the domain, number of timestamps in the data, penalization levels, dimension of discretization etc. These results might be referred to in the commentary that follows directly under this paragraph.

Now let us define more precisely the framework chosen to highlight the properties of our work. First, let's note that we have made a particular choice of acquisition model. Indeed, the acquisition operator still has the shape $\Phi : \rho \mapsto (\int_{\Omega} \phi_i(x)d\rho)_{1 \leq i \leq N}$, with the choice of function $\phi_i : x \mapsto \exp(-\|x - x_i\|^2/\sigma^2)$ with the x_i points on a $N \times N$ uniform grid. This particular choice of test functions defining our acquisition is motivated by the aim of our work towards application to dynamic deconvolution of the trajectories of contrast agents in medical imaging. It is also very accomodating as this can simply be interpreted as a discretized convolution with a PSF and it is also easy to plot.

The first two synthetic examples we will be interested in are the ones that are usually presented in the literature Duval and Tovey 2022; Bredies, Carioni, Fanzon, et al. 2022.

Gradient descents were performed using Pytorch and the associated automatic differentiation framework. To compute the exponential maps needed to define piecewise geodesic curves and to carry out the Riemannian gradient descent, we need to use numerical integration. To do so we used the torchdiffeq package R. T. Q. Chen 2018 that defines numerous integration schemes and provides automatic differentiation capabilities. We used the default DOPRI5 which is a adaptative timestep version of Runge-Kutta method at the order 5 (see Hairer et al. 2008 for more information).

For the Euclidean case we used the Adam solver with learning rate equal to $1e - 2$ to perform gradient descent. Bezier curves were computed using a naive implementation of de Casteljau's algorithm.

For every result presented here we have used a multistart strategy for each atom to be added to the solution, similarly as what is performed in Bredies, Carioni, Fanzon, et al. 2022. Each time we have sampled control points according to the density $Q(-\eta_t)$ where $Q = \exp$, we perform gradient descent for each of these curves individually and keep only the one that yields the smallest energy value.

As the reader will notice when looking at these results, we are not yet interested in estimating the number of diracs to be found or the associated

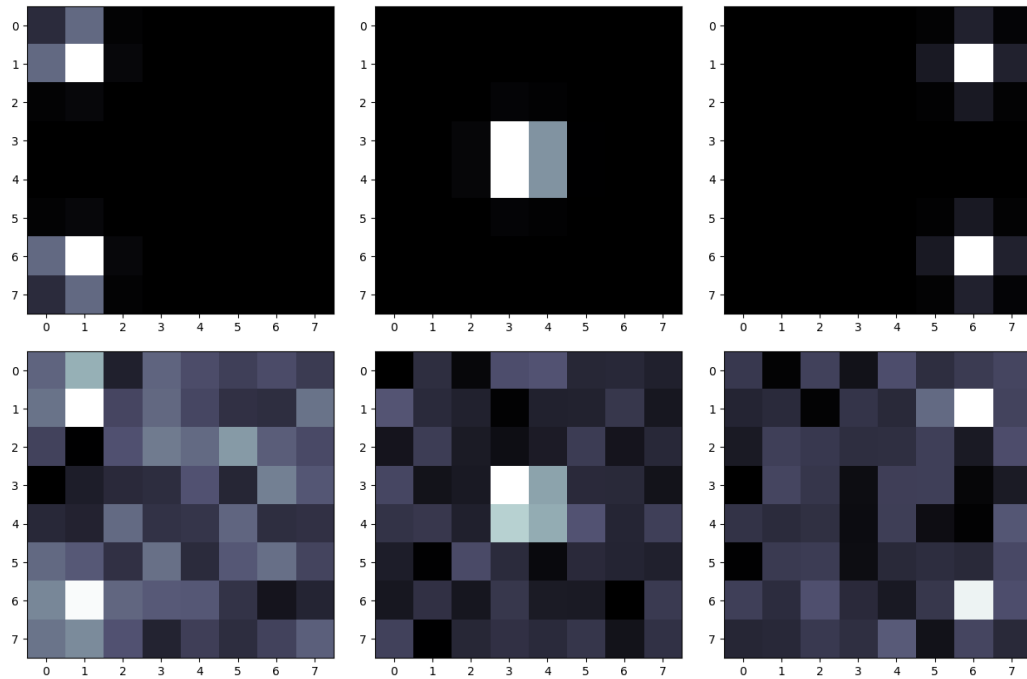


Figure 4.5: Acquisition used as an example at three time steps $t=0, 0.5, 1$. (left to right), clean (top) and with 10% noise.

Crossing curves

Our first example is the same as one of the most commonly used in the literature. It simply consists in two straight symmetric curves crossing each other in the center of the domain. The underlying curves are easily described as parametrized curves : $\gamma_0(t) = (-0.8 + 1.6t, -0.8 + 1.6t)$ and $\gamma_1(t) = (-0.8 + 1.6t, 0.8 - 1.6t)$ with time t taking values in $[0, 1]$.

For our acquisition operator we have chosen the vertices of the uniform 8×8 grid taken on the domain $[-1, 1] \times [-1, 1]$.

We set the standard deviation of the gaussian test functions in the acquisition process to $\sigma = 0.2$.

We have sampled 128 sets of values for the control points of the for each case according to the strategy described above.

We can see in Figures 4.6 and 4.7 that we can retrieve the first two atoms like in the literature. Subsection 4.2.10 further down shows tables of results for various parameters. As one can see in Table 4.1, it is in fact possible to find the two crossing curves for example when noise is much higher compared to what is shown in the Figures here and acquisition in higher dimension. We can also see that in those examples, the choice of discretization has consequences over the selected curves, and for instance Table 4.1 clearly shows that Bezier curve discretization makes it much easier to find crossing curves versus not finding them. Bezier curves are smooth curves and has some non-local properties as any point sampled inside the curve depends on all of the control points defining the curve, these properties seem to be

Figure 4.8 shows the result for the piecewise geodesic curve. The curves are

effectively untangled and a good support is recovered for both our curves. It may be noted though that the computation time is highly increased.

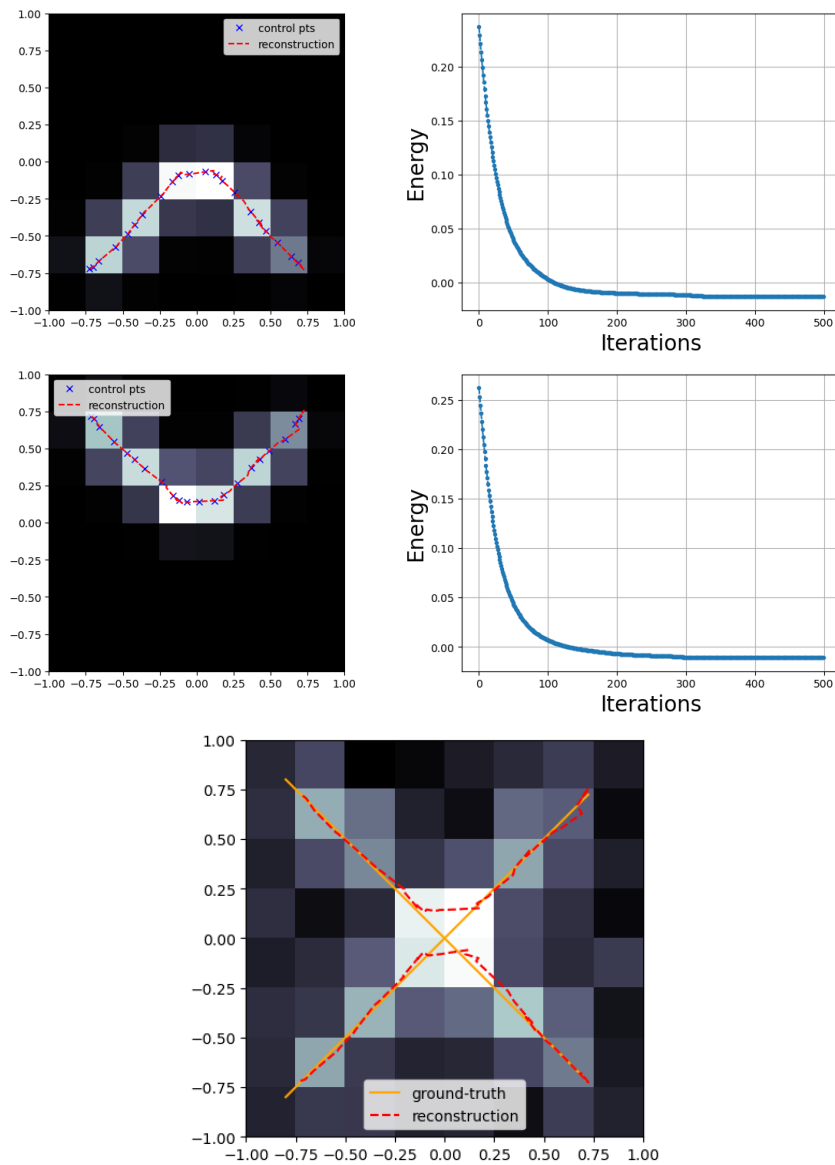


Figure 4.6: Output of our method using the Polygonal discretization of curves in the Euclidean setting with $\beta = 0.4$ and 10% noise.

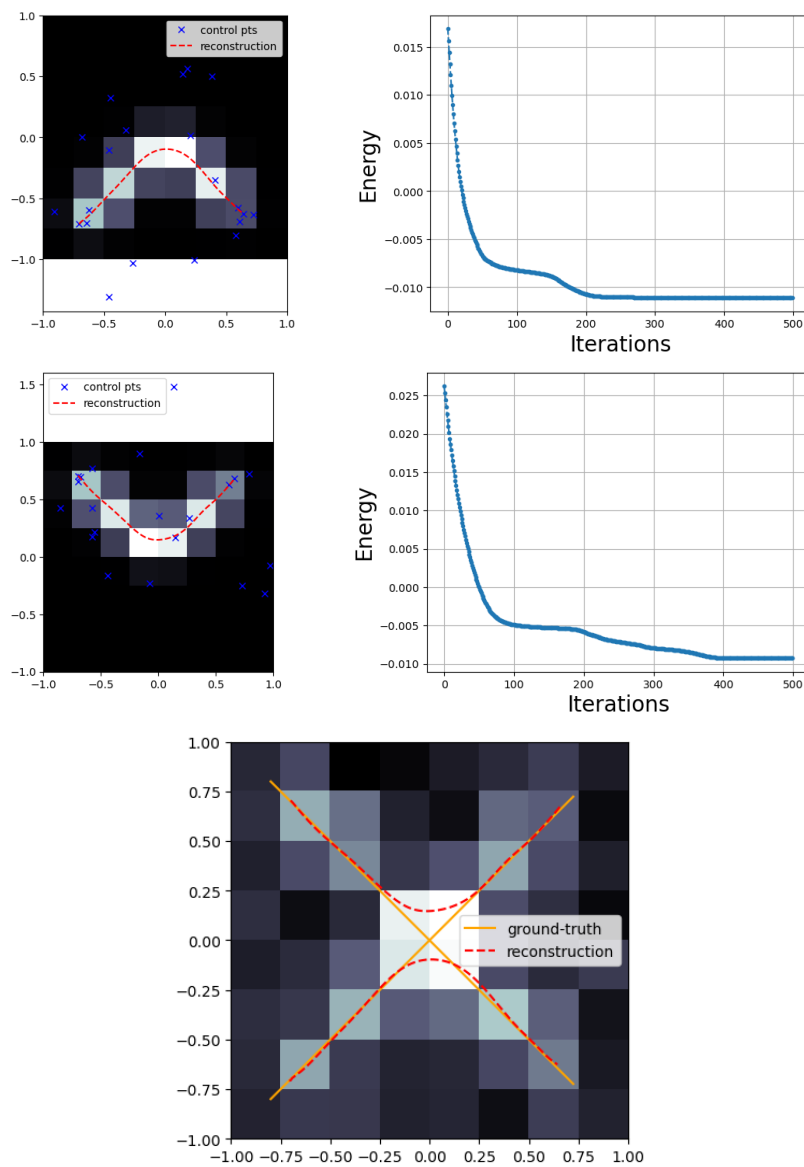


Figure 4.7: Output of our method using the Bézier discretization of curves in the Euclidean setting with $\beta = 0.4$ with 10% noise.

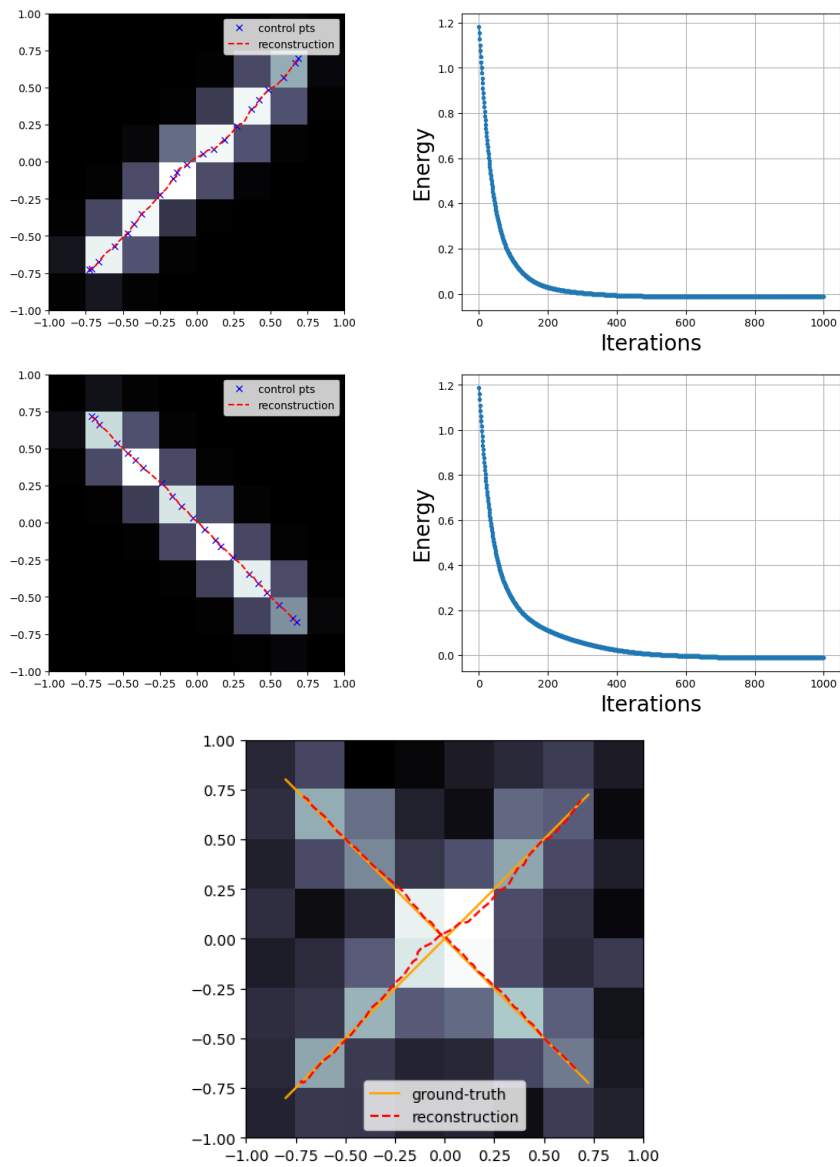


Figure 4.8: Output of our method using the Piecewise geodesic discretization of curves in the relaxed Reeds-Shepp in the Euclidean setting with $\beta = 0.4$ and 10% noise.

Three curves

Now we show our results on the second example that is commonly studied in the literature, it is composed of three curves to be recovered with one that is a fraction of a circle, a straight line and another more complex curve that is tangent to the straight line.

As one can see on Figures 4.10 and 4.11 for the case with 10% noise, 16 timesteps and a 16×16 grid domain, it is usually hard to recover the right support for the curves. We have sampled 128 sets of values for the control points of the for each case according to the strategy described above. In both of these outputs we can notice huge entangling of the supports of the different recovered curves. It is not the desired result as our motivation for this work was the ability to recover the right support, but it can be said at least that the projection $\rho_t = e_{t\sharp}\sigma$ seems to be a good approximation of the "density-velocity field" parametrization of the evolving densities. Once again, the model is highly sensitive to variations in hyperparameters and for different levels of noise and dimensions of the acquisition it is possible to recover very accurate curves as in Figure 4.13.

On this particular example, the Piecewise Geodesic model performs better as it is able to accurately separate the three curves, although the support is not completely exact as one can see on Figure 4.12.

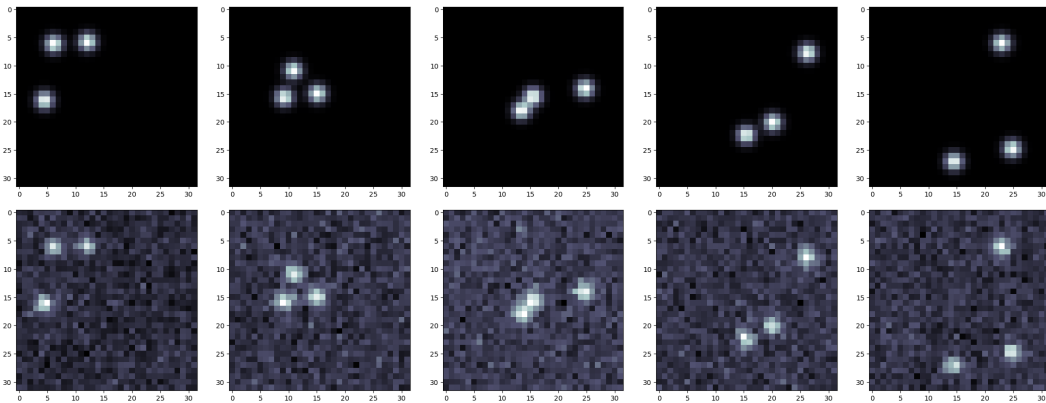


Figure 4.9: Acquisition used as an example at five timesteps $t=0, 0.25, 0.5, 0.75, 1$. (left to right), clean (top) and with 10% noise.

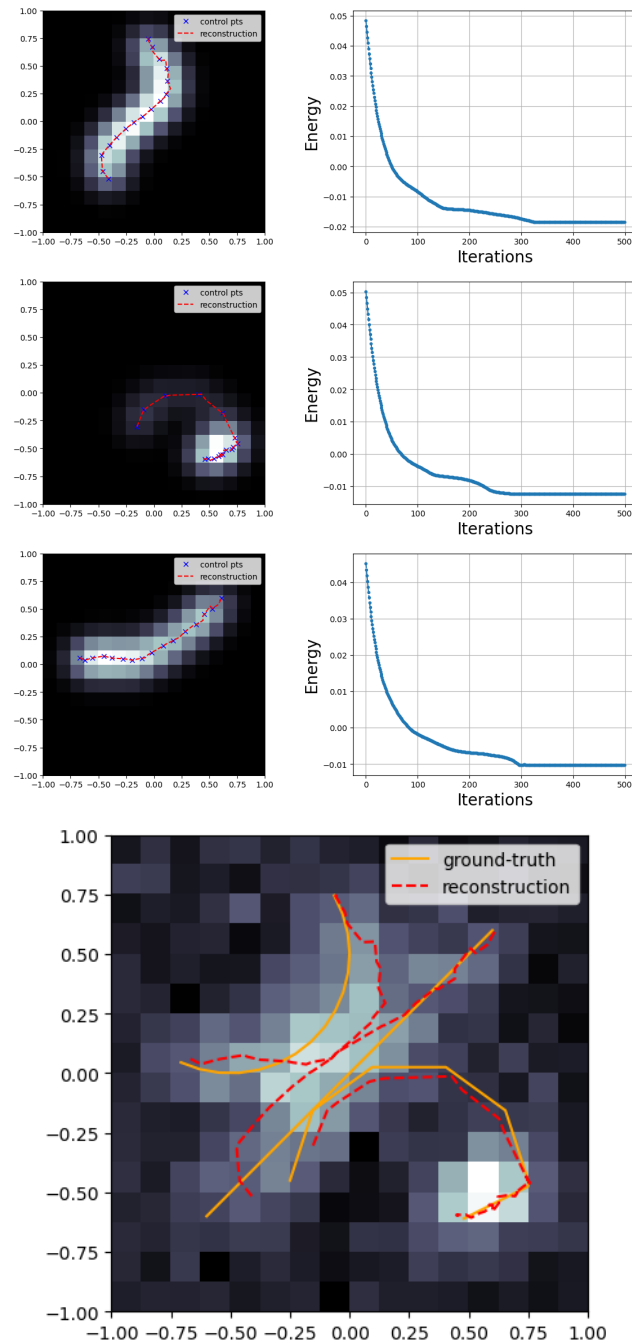


Figure 4.10: Output of our method using the Polygonal discretization of curves in the Euclidean setting with $\beta = 0.08$ and 10% noise.

4.2. Benamou-Brenier regularization for off-the-grid Curve recovery in the space of positions and on

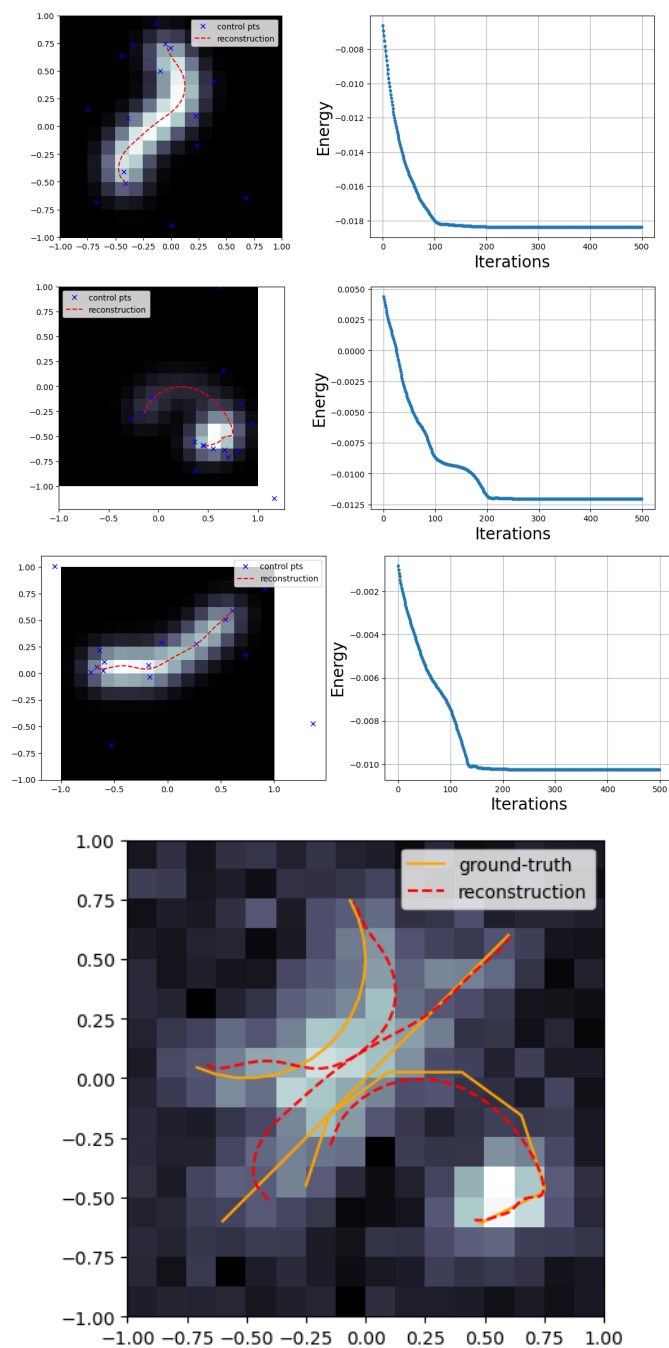


Figure 4.11: Output of our method using the Bezier discretization of curves in the Euclidean setting with $\beta = 0.05$ and 10% noise.

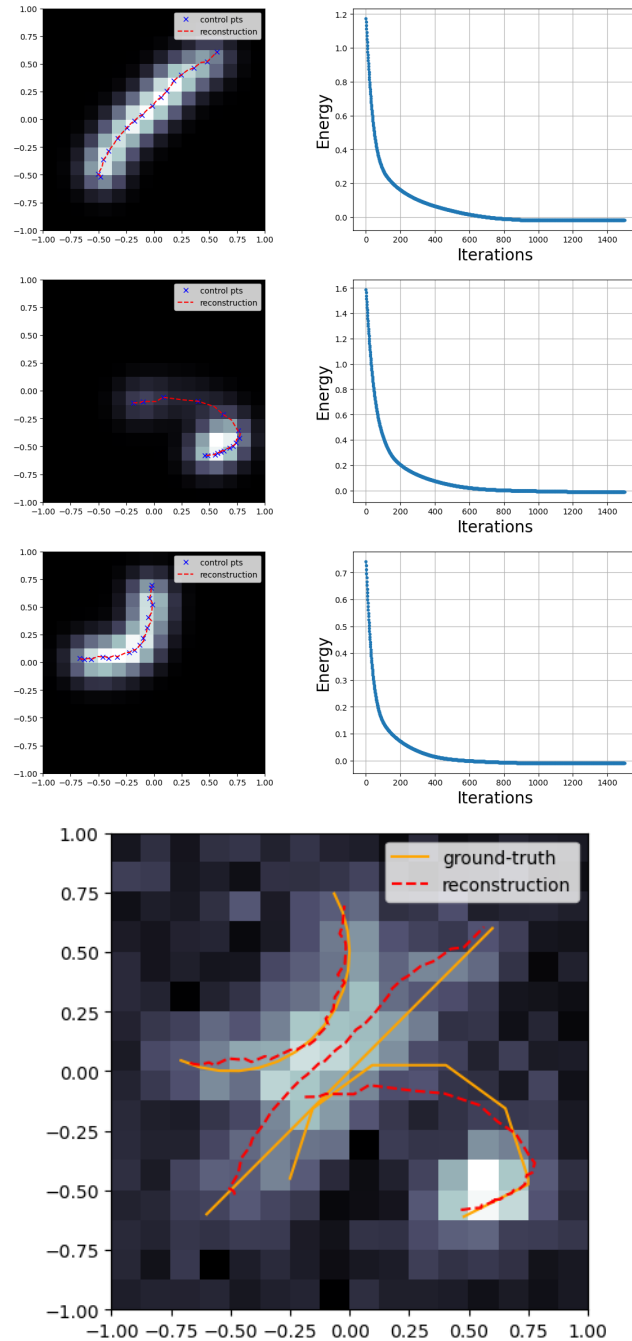


Figure 4.12: Output of our method using the Piecewise geodesic discretization of curves in the Euclidean setting with $\beta = 0.05$, $\epsilon = 0.3$, $\xi = 2$. and 10% noise.

4.2.6 Partial Conclusion and outlook

We have successfully extended the model for curve recovery with Benamou-Brenier-like regularization to the Riemannian manifold framework. We have also investigated new methods to parametrize the problem and find the curves that achieve the minimum of our problem. We have proposed a naive criterium to find such suitable discretizations.

This work is not yet complete and we still need to propose a discretization that is both suitable and *realistic for numerical computations*. We have investigated the methods in a very simple way by simply trying to look for the first atoms discovered for our models, a more accurate evaluation would also take into account the need to estimate the number of atoms in the decomposition of the minimizer. The implementation of a good stopping criterion relying on the properties of the problem is needed to be really relevant. Going forward we want to investigate the possibility to generalize Bezier curves on metric spaces, either by using de Casteljau's algorithm or parametrizing such a curve as the Fréchet mean of control points with weights depending smoothly on time. Such discretization needs further study as guaranteeing their suitability is not straightforward and will need some analytical work taking advantage of comparison theorems in Riemannian geometry in order to provide the necessary bounds on the energy.

In future works, it could be interesting to consider the extension of this method to an off-the-grid static context, namely the reconstruction (and untangling) of curves from one image Laville et al. 2023 rather than a stack. Both problems, dynamic path and static curves, share a lot of similarities and could benefit from each other improvements. Also, our proposed algorithm could be tested on real data images and delivered in a so-called '*off-the-shelf*' package, which may come handy for biologists and applicative researchers with a GUI.

4.2.7 Appendix : Extreme points of the Benamou-Brenier Energy (Riemannian case)

Here we gather a few results to ensure the relationships between the analytic and metric definitions of length, distance and (metric) derivatives in order to verify that the nature of extreme points of the regularizer doesn't change when setting the problem in manifolds. For more detail on manifolds geodesic distance and their length structure we refer to Burtscher 2015.

Lemma 4.18. *If the distance d is the Euclidean distance, and $\gamma \in \Gamma$ is differentiable, then the metric derivative corresponds to the norm of the classical (or analytic) derivative : $\forall t \in [0, 1], \quad |\dot{\gamma}|(t) = \|\gamma'(t)\|$.*

Similarly, if d is the geodesic distance associated with the metric tensor g , then the metric derivative corresponds to the metric applied to the classical derivative : $\forall t \in [0, 1], \quad |\dot{\gamma}|(t) = \sqrt{g_{\gamma(t)}(\gamma'(t), \gamma'(t))}$.

Proof. The euclidean case comes directly from the definition of the derivative, and the Riemannian case is proven via a short computation :

$$\begin{aligned} \frac{d(\gamma(t), \gamma(t + \eta))}{|\eta|} &= \frac{\sqrt{g_{\gamma(t)}(\text{Log}_{\gamma(t)}(\gamma(t + \eta)), \text{Log}_{\gamma(t)}(\gamma(t + \eta)))}}{|\eta|} \\ &= \sqrt{g_{\gamma(t)}\left(\frac{\text{Log}_{\gamma(t)}(\gamma(t + \eta)) - \text{Log}_{\gamma(t)}(\gamma(t))}{\eta}, \frac{\text{Log}_{\gamma(t)}(\gamma(t + \eta)) - \text{Log}_{\gamma(t)}(\gamma(t))}{\eta}\right)} \\ &\xrightarrow{\eta \rightarrow 0} \sqrt{g_{\gamma(t)}(d\text{Log}_{\gamma(t)}\gamma(t) \cdot \gamma'(t), d\text{Log}_{\gamma(t)}\gamma(t) \cdot \gamma'(t))} = \sqrt{g_{\gamma(t)}(\gamma'(t), \gamma'(t))} \end{aligned}$$

□

Lemma 4.19. *The functional $w : \gamma \in \Gamma^{\mathcal{M}} \mapsto \int_0^1 \alpha + \beta \|\gamma'(t)\|_g dt$, with $g \in \mathcal{C}^2(\mathcal{M})$, is lower semi-continuous.*

Proof. w may be rewritten as the supremum of a family of continuous functions of γ :

$$w(\gamma) = \sup_{t_0=0 < t_1 < \dots < t_n=1} \alpha + \beta \sum_{k=0}^{n-1} \frac{d_g(\gamma(t_k), \gamma(t_{k+1}))}{t_{k+1} - t_k},$$

thus ensuring lower semi-continuity of w .

□

Lemma 4.20. *Let $w : \Gamma \rightarrow]0, +\infty]$ a lower semi-continuous function and $D = \left\{ \sigma \in \mathcal{M}(\Gamma) / \int_{\Gamma} w d\sigma \leq 1 \right\}$. D is closed and:*

$$\text{Ext}(D) = \{0\} \cup \{w(\gamma)^{-1} \delta_{\gamma} | w(\gamma) < +\infty\}$$

Proof. For any convex decomposition $\sigma = \lambda\sigma_1 + (1 - \lambda)\sigma_2$, if $\sigma = w(\gamma)^{-1}\delta_\gamma$ for any $\gamma \in \Gamma$, then σ_1 and σ_2 are supported only on γ , $\sigma_1 = \frac{\alpha}{w(\gamma)}\delta_\gamma$, $\sigma_2 = \frac{\beta}{w(\gamma)}\delta_\gamma$, $\alpha, \beta \in \mathbb{R}_+$.

And then, as $\int w d\sigma = 1 = \lambda\alpha + (1 - \lambda)\beta$ and $\alpha + \beta = 1$, we have $\alpha = \beta = 1$ and $\sigma = \sigma_1 = \sigma_2$. In the end, $w(\gamma)^{-1}\delta_\gamma$ is indeed an extremal point of D .

Let $\sigma \in \text{Ext}(D)$, and make the assumption that there are at least 2 curves γ_0 and γ_1 in the support of σ . Let $\Gamma_0 = \{d_\Gamma(\gamma, \gamma_0) \leq \frac{1}{2}d_\Gamma(\gamma_1, \gamma_0)\}$, and $\Gamma_1 = \Gamma_0^c$, defining $\sigma_0 = \mathbb{1}_{\Gamma_0}\sigma$ and $\sigma_1 = \sigma - \sigma_0$.

If $\int w d\sigma_i \neq 0$ and $\int w d\sigma_i \leq \int w d\sigma \leq 1$, then with $\lambda = \int w d\sigma_0$, we get that $\sigma = \lambda \frac{\sigma_0}{\lambda} + (1 - \lambda) \frac{\sigma_1}{1 - \lambda}$ and we have written σ as a convex combination of elements of D , so either $\sigma \notin \text{Ext}(D)$ or for a certain $i \in \{0, 1\}$, $\int w d\sigma_i = 0$. In the last case we have that by positivity of w , $\sigma = 0$ and then we can't have $\gamma_i \in \Gamma_i$, which contradicts the assumption.

In the end, we have that σ has at most 1 curve in its support which means either $\sigma = 0$ or $\sigma = \frac{1}{w(\gamma)}\delta_\gamma$ for some $\gamma \in \Gamma$. \square

4.2.8 Appendix : Computations for the relaxed Reeds-Shepp

With the help of the SageMath software, we carry out explicit computations of the christoffel symbols for the relaxed Reeds-Shepp metric model.

Let $p = (x, y, \theta) \in \mathbb{M}_2$ a rototranslation, the metric tensor g at p is defined by:

$$g_{ij} = \begin{bmatrix} \cos(\theta)^2 + \varepsilon^{-2} \sin(\theta)^2 & (1 - \varepsilon^{-2}) \cos(\theta) \sin(\theta) & 0 \\ (1 - \varepsilon^{-2}) \cos(\theta) \sin(\theta) & \sin(\theta)^2 + \varepsilon^{-2} \cos(\theta)^2 & 0 \\ 0 & 0 & \xi^2 \end{bmatrix},$$

with constants $\varepsilon, \xi > 0$ while its inverse reads:

$$g^{ij} = \begin{bmatrix} \cos(\theta)^2 + \varepsilon^2 \sin(\theta)^2 & (1 - \varepsilon^2) \cos(\theta) \sin(\theta) & 0 \\ (1 - \varepsilon^2) \cos(\theta) \sin(\theta) & \sin(\theta)^2 + \varepsilon^2 \cos(\theta)^2 & 0 \\ 0 & 0 & \xi^{-2} \end{bmatrix}.$$

The Christoffel symbols, up to the symmetries² $\Gamma_{ij}^k = \Gamma_{ji}^k$ and the null symbols,

²The Levi-Civita connection ∇ is *by definition* torsion-free.

of the Levi-Civita connection ∇ of g write down:

$$\begin{aligned}\Gamma_{x,\theta}^x &= -\frac{1}{2\varepsilon^2}(\varepsilon^4 - 1) \cos(\theta) \sin(\theta) \\ \Gamma_{y,\theta}^x &= -\frac{1}{2\varepsilon^2}(\varepsilon^4 - (\varepsilon^4 - 1) \cos(\theta)^2 - \varepsilon^2) \\ \Gamma_{x,\theta}^y &= \frac{1}{2\varepsilon^2}((\varepsilon^4 - 1) \cos(\theta)^2 - \varepsilon^2 + 1) \\ \Gamma_{y,\theta}^y &= \frac{\varepsilon^4 - 1}{2\varepsilon^2} \cos(\theta) \sin(\theta) \\ \Gamma_{x,x}^\theta &= \frac{\varepsilon^2 - 1}{\varepsilon^2 \xi^2} \cos(\theta) \sin(\theta) \\ \Gamma_{x,y}^\theta &= -\frac{1}{2\varepsilon^2 \xi^2} (2(\varepsilon^2 - 1) \cos(\theta)^2 - \varepsilon^2 + 1) \\ \Gamma_{y,y}^\theta &= -\frac{\varepsilon^2 - 1}{\varepsilon^2 \xi^2} \cos(\theta) \sin(\theta).\end{aligned}$$

4.2.9 Appendix : Riemannian optimization on Manifold and Tangent bundle

For first order optimization, there is a simple adaptation of gradient descent on Riemannian manifold. The Riemannian gradient descent simply consists in a gradient descent, but we take into account the non-Euclidean nature of the manifold by leveraging the Riemannian exponential map to update the elements of the minimizing sequence. The procedure is described in the Algorithm 4.

Algorithm 5 : Riemannian gradient descent

Data : Objective function $f : x \in \mathcal{M} \mapsto f(x) \in \mathbb{R}$, initialisation point $x_0 \in \mathcal{M}$, number of timesteps $T \in \mathbb{N}^*$

1 **for** $t = 0, 1, 2, \dots, T - 1$ **do**

2 Compute the Riemannian gradient $\nabla_{\mathcal{M}} f(x)$.

3 Update the iterate by:

$$x = \text{Exp}_x(-\eta \nabla_{\mathcal{M}} f(x))$$

 where Exp_x denotes the Riemannian exponential map at point x .

4 **end**

Result : Point x of the manifold

Usually, we replace the use of the exponential map by a *retraction* i.e. a first order approximation of this mapping. Now we want to carry out Riemannian gradient descent, not only on the base manifold itself but on the tangent bundle, which means the objective function is now a map $f : (x, v) \in \mathcal{M} \times T_x \mathcal{M} \mapsto f(x, v) \in \mathbb{R}$.

The tangent bundle $T\mathcal{M}$ is itself a $2d$ -dimensional manifold with tangent bundle $TT\mathcal{M}$. For any base point $(p, u) \in T\mathcal{M}$, $T_{(p,u)}T\mathcal{M}$ has a particular structure as it can be decomposed into Horizontal and Vertical subspaces $T_{(p,u)}T\mathcal{M} =$

$\mathcal{H}_{(p,u)} \oplus \mathcal{V}_{(p,u)}$, both d -dimensional. There is a natural lift from $T_p\mathcal{M}$ to the Horizontal and Vertical subspaces, let $X \in T_p\mathcal{M}$ we write X^h the horizontal part and X^v the vertical part of X .

A natural choice of metric for the Tangent bundle manifold is called the Sasaki metric. It simply consists in using the base metric and imposing perpendicularity of the Horizontal and Vertical subspaces :

1. $\hat{g}_{(p,u)}(X^h, Y^h) = g_p(X, Y)$,
2. $\hat{g}_{(p,u)}(X^v, Y^v) = g_p(X, Y)$,
3. $\hat{g}_{(p,u)}(X^h, Y^v) = 0$,

Computing the Riemannian exponential might be hard or inefficient, as we usually don't have access to any closed form formulae. *Retractions* are a good way to circumvent the difficulty of computing the Riemannian exponential map. On the Sasaki manifold $(T\mathcal{M}, \hat{g})$ we will define the map

$$R_{(p,u)}((v, w)) = (\text{Exp}_p v, \Xi_p^{\text{Exp}_p v}(u + w)).$$

This map is a retraction over the Sasaki manifold Rentmeesters 2011. It is to be interpreted as simply : point p is sent following the geodesic defined

Replacing the exponential map by a retraction R in the gradient descent algorithm

Algorithm 6 : Retraction gradient descent on the tangent bundle

Data : Objective function $f : (x, v) \in \mathcal{M} \mapsto f(x, v) \in \mathbb{R}$, initialisation
point $(x_0, v_0) \in T\mathcal{M}$, number of timesteps $T \in \mathbb{N}^*$

1 **for** $t = 0, 1, 2, \dots, T - 1$ **do**

2 Compute the Riemannian gradient $\nabla_{T\mathcal{M}} f(x, v)$.

3 Update the iterate by:

$$(x^{k+1}, v^{k+1}) = R_{(x^k, v^k)}(-\eta \nabla_{T\mathcal{M}} f(x^k, v^k))$$

where R_x denotes the Riemannian exponential map at point x .

4 **end**

Result : Point (x^T, v^T) on the tangent bundle of the base manifold \mathcal{M} .

Energy in the piecewise geodesic framework on the Sasaki metric

For ease of implementation we consider, instead of the straightforward piecewise geodesic model, a set of points on the Sasaki Manifold associated with the base Reeds-Shepp manifold $(x_i, v_i)_{1 \leq i \leq n-1}$. The piecewise geodesic curve interpolating the points (x_i) corresponds to the condition $\text{Exp}_{x_i} v_i = x_{i+1} \forall 1 \leq i \leq n - 1$.

We bring slight modifications of the energy to be optimized by simply adding a penalization to the unconstrained problem in order to avoid having to compute the Riemannian Log map which would be very expensive.

In the end, the energy we minimize in the Oracle step is the following :

$$F((x_i, v_i)) = \sum_i \langle \delta_{\gamma(x_i, v_i)}(t_i), \eta^{[k]} \rangle + \beta \sum_i (t_{i+1} - t_i) \|v_i\|_{x_i}^2 + \lambda \sum_i \|\text{Exp}_{x_i} v_i - x_{i+1}\|^2,$$

If we have the same number of timestamps and points defining the curve piecewise geodesic curve, the objective becomes :

$$F((x_i, v_i)) = \sum_i \eta^{[k]}(x_i) + \beta \sum_i (t_{i+1} - t_i) \|v_i\|_{x_i}^2 + \lambda \sum_i \|\text{Exp}_{x_i} v_i - x_{i+1}\|^2,$$

4.2.10 Appendix : Numerical results in the detail

In this subsection we present tables showing the results of the implemented methods presented above. We have performed a wide variety of experiments for different sets of parameters and also varying the setup of our problem in terms of dimension of the acquisition (by changing the size of the grid covering the domain) and level of noise. In the following tables we evaluate our result in terms of the distance between the recovered curves and the ground truth. Specifically we evaluate

$$\sum_i \min_j (\|\gamma_i - \tilde{\gamma}_j\|_{L^2(0,1)}^2).$$

Notice that, for the case of the two crossing curves, it is very easy to evaluate when we have been able to accurately recover the support of both curves as the value of the evaluation is one or two orders of magnitude lower.

For more clarity let us be very explicit on the meaning of each of the variable names for each column in the following tables :

- **Method** : name of the parametrization used,
- **epochs** : number of iterations in the gradient descent scheme,
- **lr** : learning rate or rather the size of the steps used in the gradient descent scheme,
- **noise** : noise level in the input,
- **n** : number of timestamps in the input,
- **nc** : number of control points for the parametrization of the curves,
- **n_start** : number of curves used for multistart,
- **regularization** : value of the parameter β ,
- **nb_pics** : number of curves to recover,
- **epsilon** and **xi** : parameters in the Reeds-Shepp metric model,
- **Runtime** : total run time in seconds to carry out the gradient descents to find the set number of curves,

4.2. Benamou-Brenier regularization for off-the-grid Curve recovery in the space of positions and on

- **evaluation** : value of the squared euclidean distance computed between the output and the ground truth curves, as described above.

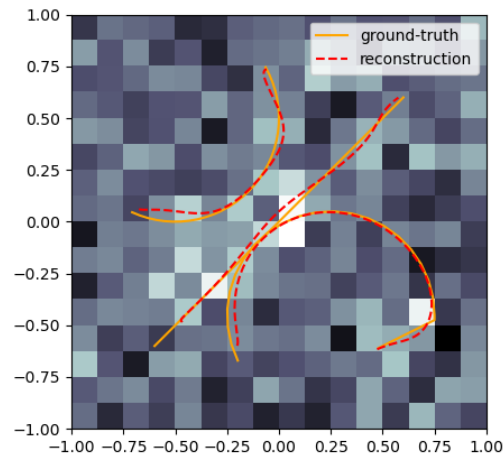


Figure 4.13: Reconstruction corresponding to the best result (for the evaluation metric) shown in Table 4.2.

| Method | epochs | lr | noise | n | nc | n_start | regularization | Runtime (s) | evaluation |
|-----------|--------|------|-------|----|----|---------|----------------|-------------|------------|
| bezier | 500 | 0.01 | 0.6 | 32 | 8 | 64 | 0.01 | 49 | 0.0024 |
| bezier | 500 | 0.01 | 0.6 | 32 | 16 | 8 | 0.1 | 22 | 0.0026 |
| bezier | 500 | 0.01 | 0.6 | 32 | 16 | 64 | 0.1 | 55 | 0.0026 |
| bezier | 500 | 0.01 | 0.6 | 32 | 16 | 64 | 0.05 | 52 | 0.0028 |
| bezier | 500 | 0.01 | 0.6 | 32 | 16 | 8 | 0.05 | 21 | 0.0029 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 8 | 8 | 0.05 | 16 | 0.0032 |
| bezier | 500 | 0.01 | 0.6 | 32 | 32 | 64 | 0.1 | 75 | 0.0033 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 16 | 64 | 0.1 | 39 | 0.0035 |
| bezier | 500 | 0.01 | 0.6 | 32 | 4 | 64 | 0.01 | 57 | 0.0035 |
| bezier | 500 | 0.01 | 0.6 | 32 | 4 | 8 | 0.01 | 21 | 0.0035 |
| bezier | 500 | 0.01 | 0.6 | 32 | 32 | 8 | 0.05 | 29 | 0.0038 |
| bezier | 500 | 0.01 | 0.6 | 32 | 32 | 64 | 0.05 | 74 | 0.0038 |
| bezier | 500 | 0.01 | 0.6 | 32 | 16 | 64 | 0.01 | 56 | 0.0038 |
| bezier | 500 | 0.01 | 0.6 | 32 | 16 | 8 | 0.01 | 21 | 0.0038 |
| bezier | 500 | 0.01 | 0.6 | 32 | 32 | 64 | 0.01 | 77 | 0.0106 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 4 | 8 | 0.01 | 16 | 0.0151 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 4 | 64 | 0.01 | 42 | 0.0174 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 4 | 64 | 0.05 | 43 | 0.0304 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 16 | 8 | 0.05 | 16 | 0.0328 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 32 | 64 | 0.05 | 40 | 0.1269 |
| bezier | 500 | 0.01 | 0.6 | 32 | 64 | 8 | 0.1 | 46 | 0.2102 |
| bezier | 500 | 0.01 | 0.6 | 32 | 64 | 8 | 0.05 | 50 | 0.2387 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 32 | 8 | 0.1 | 16 | 0.2859 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 16 | 8 | 0.1 | 16 | 0.3092 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 4 | 8 | 0.05 | 16 | 0.3141 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 4 | 8 | 0.1 | 17 | 0.3229 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 4 | 64 | 0.1 | 40 | 0.3230 |
| bezier | 500 | 0.01 | 0.6 | 32 | 4 | 64 | 0.05 | 48 | 0.3342 |
| bezier | 500 | 0.01 | 0.6 | 32 | 4 | 8 | 0.05 | 17 | 0.3343 |
| bezier | 500 | 0.01 | 0.6 | 32 | 64 | 64 | 0.05 | 156 | 0.3403 |
| bezier | 500 | 0.01 | 0.6 | 32 | 64 | 64 | 0.01 | 151 | 0.3459 |
| bezier | 500 | 0.01 | 0.6 | 32 | 8 | 8 | 0.1 | 18 | 0.3608 |
| bezier | 500 | 0.01 | 0.6 | 32 | 8 | 64 | 0.1 | 46 | 0.3612 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 32 | 8 | 0.05 | 16 | 0.3710 |
| bezier | 500 | 0.01 | 0.6 | 32 | 8 | 8 | 0.05 | 18 | 0.3782 |
| bezier | 500 | 0.01 | 0.6 | 32 | 8 | 64 | 0.05 | 59 | 0.3782 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 8 | 64 | 0.1 | 41 | 0.3791 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 8 | 8 | 0.1 | 17 | 0.3791 |
| bezier | 500 | 0.01 | 0.6 | 32 | 4 | 64 | 0.1 | 42 | 0.3838 |
| bezier | 500 | 0.01 | 0.6 | 32 | 4 | 8 | 0.1 | 20 | 0.3838 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 8 | 64 | 0.05 | 39 | 0.3875 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 8 | 8 | 0.01 | 16 | 0.3902 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 8 | 64 | 0.01 | 40 | 0.3903 |
| bezier | 500 | 0.01 | 0.6 | 32 | 8 | 8 | 0.01 | 18 | 0.3938 |
| bezier | 500 | 0.01 | 0.6 | 32 | 32 | 8 | 0.1 | 30 | 0.3960 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 16 | 8 | 0.01 | 16 | 0.3963 |
| bezier | 500 | 0.01 | 0.6 | 32 | 64 | 8 | 0.01 | 52 | 0.4020 |
| bezier | 500 | 0.01 | 0.6 | 32 | 64 | 64 | 0.1 | 148 | 0.4051 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 16 | 64 | 0.05 | 42 | 0.4070 |
| bezier | 500 | 0.01 | 0.6 | 32 | 32 | 8 | 0.01 | 30 | 0.4111 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 16 | 64 | 0.01 | 41 | 0.4177 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 32 | 64 | 0.1 | 42 | 0.4280 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 64 | 64 | 0.1 | 39 | 0.4548 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 64 | 8 | 0.1 | 16 | 0.5003 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 64 | 64 | 0.05 | 41 | 0.5549 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 32 | 64 | 0.01 | 41 | 0.5712 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 32 | 8 | 0.01 | 16 | 0.6063 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 64 | 8 | 0.05 | 16 | 0.6206 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 64 | 8 | 0.01 | 17 | 0.6632 |
| polygonal | 500 | 0.01 | 0.6 | 32 | 64 | 64 | 0.01 | 42 | 0.7006 |

Table 4.1: Results in euclidean geometry for multiple sets of hyperparameters for two crossing curves to recover and 60% noise, n=32

4.2. Benamou-Brenier regularization for off-the-grid Curve recovery in the space of positions and on

| method | epochs | lr | noise | n | n_start | nb_pics | nc | regularization | Runtime(s) | evaluation |
|-----------|--------|------|-------|----|---------|---------|----|----------------|------------|------------|
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 8 | 0.01 | 91 | 0.0019 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 8 | 0.05 | 69 | 0.0720 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 8 | 0.05 | 91 | 0.1344 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 32 | 0.1 | 143 | 0.1603 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 16 | 0.1 | 72 | 0.1867 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 16 | 0.05 | 83 | 0.1928 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 8 | 0.01 | 64 | 0.1991 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 32 | 0.01 | 279 | 0.2036 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 16 | 0.05 | 57 | 0.2038 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 64 | 0.1 | 592 | 0.2571 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 64 | 0.05 | 638 | 0.2731 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 64 | 0.01 | 623 | 0.2736 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 32 | 0.05 | 141 | 0.2779 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 8 | 0.1 | 56 | 0.2811 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 16 | 0.01 | 82 | 0.3040 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 16 | 0.1 | 129 | 0.3190 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 32 | 0.1 | 66 | 0.3390 |
| bezier | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 8 | 0.1 | 91 | 0.3610 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 64 | 0.1 | 62 | 0.4633 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 64 | 0.05 | 62 | 0.4870 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 16 | 0.01 | 66 | 0.5154 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 64 | 0.01 | 62 | 0.7673 |
| polygonal | 500 | 0.01 | 0.4 | 32 | 64 | 3 | 32 | 0.01 | 65 | 0.8804 |

Table 4.2: Results in the euclidean setting for 3 curves to recover with various parameters, with 40% noise, on a 16×16 grid domain and 32 timestamps.

| method | nb_pics | n | noise | epochs | n_start | lr | nc | regularization | epsilon | xi | Runtime (s) | evaluation |
|----------------|---------|----|-------|--------|---------|------|----|----------------|---------|-----|-------------|------------|
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.1 | 0.3 | 2 | 2332 | 0.0089 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.1 | 0.3 | 1 | 2274 | 0.0357 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.1 | 0.3 | 2 | 3363 | 0.1284 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.05 | 0.3 | 1 | 4724 | 0.1297 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.01 | 0.3 | 2 | 4995 | 0.1620 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.01 | 0.3 | 0.5 | 18934 | 0.2842 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.05 | 0.3 | 2 | 2942 | 0.3299 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.1 | 0.3 | 1 | 3384 | 0.3461 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.005 | 0.3 | 0.5 | 22878 | 0.3467 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.05 | 0.3 | 0.5 | 25217 | 0.3559 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.1 | 0.3 | 0.5 | 9284 | 0.3621 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.05 | 0.3 | 0.5 | 11015 | 0.3642 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.1 | 0.3 | 0.5 | 17279 | 0.3646 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.01 | 0.3 | 1 | 8323 | 0.3673 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.05 | 0.3 | 1 | 3438 | 0.3695 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.005 | 0.3 | 2 | 8724 | 0.3704 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.01 | 0.3 | 1 | 10102 | 0.3725 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.1 | 0.3 | 0.5 | 12294 | 0.3816 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.05 | 0.3 | 0.5 | 17566 | 0.3934 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.05 | 0.3 | 1 | 6893 | 0.3983 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.005 | 0.3 | 1 | 13867 | 0.4118 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.005 | 0.3 | 0.5 | 31475 | 0.4133 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.1 | 0.3 | 2 | 5242 | 0.4304 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.01 | 0.3 | 0.5 | 27784 | 0.4525 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.005 | 0.3 | 2 | 6396 | 0.4590 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.005 | 0.3 | 1 | 9231 | 0.4677 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.05 | 0.3 | 2 | 4412 | 0.5189 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.01 | 0.3 | 2 | 7521 | 0.5400 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.01 | 0.3 | 1 | 12569 | 0.5697 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.05 | 0.3 | 2 | 6358 | 0.5807 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.005 | 0.3 | 0.5 | 41941 | 0.5915 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.1 | 0.3 | 1 | 4711 | 0.6270 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.01 | 0.3 | 2 | 8616 | 0.6746 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.005 | 0.3 | 2 | 12949 | 0.7008 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.005 | 0.3 | 1 | 17875 | 0.7184 |
| exponential_RS | 2 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.01 | 0.3 | 0.5 | 41185 | 0.7366 |

Table 4.3: Results for the case with 2 crossing curves to recover, with 40% noise on a 16×16 grid domain and 16 timestamps.

| method | nb_pics | n | noise | epochs | n_start | lr | nc | regularization | epsilon | xi | Runtime (s) | evaluation |
|----------------|---------|----|-------|--------|---------|------|----|----------------|---------|-----|-------------|------------|
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.05 | 0.3 | 2 | 3946 | 0.0749 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.1 | 0.3 | 2 | 3425 | 0.0822 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.1 | 0.3 | 0.5 | 12559 | 0.0986 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.05 | 0.3 | 0.5 | 17024 | 0.1009 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.05 | 0.3 | 1 | 5407 | 0.1279 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.1 | 0.3 | 1 | 3408 | 0.1566 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.1 | 0.3 | 1 | 5571 | 0.1720 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.05 | 0.3 | 0.5 | 25187 | 0.2482 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.005 | 0.3 | 2 | 9419 | 0.2537 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.01 | 0.3 | 1 | 11225 | 0.2586 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.1 | 0.3 | 2 | 8011 | 0.2594 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.1 | 0.3 | 0.5 | 17091 | 0.2960 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.01 | 0.3 | 0.5 | 29425 | 0.3035 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.1 | 0.3 | 1 | 7402 | 0.3077 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.1 | 0.3 | 2 | 5175 | 0.3365 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.05 | 0.3 | 2 | 6305 | 0.3488 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.05 | 0.3 | 1 | 7914 | 0.3586 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.01 | 0.3 | 2 | 6679 | 0.3600 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.005 | 0.3 | 0.5 | 30225 | 0.3658 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.1 | 0.3 | 0.5 | 22640 | 0.3780 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 16 | 0.005 | 0.3 | 1 | 14377 | 0.3994 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.01 | 0.3 | 2 | 9666 | 0.4394 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 64 | 0.05 | 0.3 | 2 | 9298 | 0.4420 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.005 | 0.3 | 0.5 | 44748 | 0.4580 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.01 | 0.3 | 1 | 16640 | 0.4661 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.005 | 0.3 | 1 | 19784 | 0.4916 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.01 | 0.3 | 0.5 | 47193 | 0.5502 |
| exponential_RS | 3 | 16 | 0.4 | 1500 | 128 | 0.01 | 32 | 0.005 | 0.3 | 2 | 12620 | 0.6930 |

Table 4.4: Results for the case with 3 curves to recover and Piecewise geodesic parametrization in the Reeds-Shepp geometry, 40% noise, 32×32 grid domain and 16 timestamps.

Conclusion and Future Works

The work presented in this thesis has been the opportunity to journey through a wide variety of different fields of Applied Mathematics and Computer Science. Our aim was to work at the intersection of at least two of them : Geodesic and Variational models on the one hand, and Machine Learning on the other. With this in mind we have been able to make modest contributions to the state-of-the-art of Computer Vision applied to medical images. Efficient models are becoming more and more attainable as automatic differentiation is now very accessible through efficient libraries and the interest in gathering data into well annotated datasets has been understood by the medical sector (although it is often still very expensive).

Ultrasound Localization Microscopy is a rising imaging technique that shows promising results on imaging the vascular network which could in turn allow for quick diagnostic of stroke and help save lives. Although it has now been around for roughly ten years, its is still almost uncharted territory for applied mathematicians, and it still needs more annotated data out for the Machine Learning community to get invested in the domain.

Geodesic methods on the other hand, although dating back at least to the 90's for its applications in Computer Vision, are rarely regarded as a good option for the segmentation of medical images. Our first two works show that they are capable of being incorporated in Machine Learning workflows in order to add knowledge and structure to segmentation tasks. Future works might include finding a way to fairly compare results of geodesic tracking of vascular structures to segmentation output masks from neural networks. Also, including this tracking in a larger Machine Learning pipeline might yield interesting results as it did for the segmentation of tumours with geodesic balls, though this would require to be able to properly compute a gradient for geodesic curves during the automatic differentiation. Looking at the literature involving the computation of Jacobi fields may be a first step to carry out optimization involving directly the points along geodesics.

As to Variational Methods, the last contribution presented here shows that they can still help build the future of imaging and allow great versatility as well as reliability, by mean of theoretical results on the solutions of the variational problems.

Bibliography

- Abbasi-Sureshjani, S., I. Smit-Ockeloen, J. Zhang, and B. Romeny (July 2015). “Biologically-Inspired Supervised Vasculature Segmentation in SLO Retinal Fundus Images”. In: pp. 325–334.
- Ambrosio, L., E. Brué, and D. Semola (2021). *Lectures on Optimal Transport*. UNITEXT. Springer International Publishing.
- Ambrosio, L., N. Gigli, and G. Savare (2008). *Gradient Flows: In Metric Spaces and in the Space of Probability Measures*. Lectures in Mathematics. ETH Zürich. Birkhäuser Basel.
- Ambrosio, L. and T. Rajala (2014). “Slopes of Kantorovich potentials and existence of optimal transport maps in metric measure spaces”. In: *Annali di Matematica Pura ed Applicata* 193, pp. 71–87.
- Ambrosio, L. and S. Rigot (2004). “Optimal mass transportation in the Heisenberg group”. In: *Journal of Functional Analysis* 208.2, pp. 261–301.
- Ardon, R. and L. D. Cohen (Aug. 2006). “Fast Constrained Surface Extraction by Minimal Paths”. en. In: *International Journal of Computer Vision* 69.1, pp. 127–136.
- Atkinson, K. and W. Han (2009). *Theoretical Numerical Analysis: A Functional Analysis Framework*. Texts in Applied Mathematics. Springer New York.
- Bardi, M. and I. Capuzzo-Dolcetta (1997). *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. en. Boston, MA: Birkhäuser.
- Bekkers, E. J., R. Duits, A. Mashtakov, and G. R. Sanguinetti (2015). “A PDE Approach to Data-Driven Sub-Riemannian Geodesics in SE(2)”. In: *SIAM Journal on Imaging Sciences* 8.4, pp. 2740–2770. eprint: <https://doi.org/10.1137/15M1018460>.
- Bekkers, E., R. Duits, T. Berendschot, and B. ter Haar Romeny (Jan. 2014). “A Multi-Orientation Analysis Approach to Retinal Vessel Tracking”. In: *Journal of Mathematical Imaging and Vision* 49.3, pp. 583–610.
- Bekkers, E. J., D. Chen, and J. M. Portegies (July 2018). “Nilpotent Approximations of Sub-Riemannian Distances for Fast Perceptual Grouping of Blood Vessels in 2D and 3D”. In: *Journal of Mathematical Imaging and Vision* 60.6, pp. 882–899.
- Benamou, J.-D. and Y. Brenier (2000). “A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem”. In: *Numerische Mathematik* 84, pp. 375–393.
- Benmansour, F., G. Carlier, G. Peyré, and F. Santambrogio (2010). “Derivatives with Respect to Metrics and Applications: Subgradient Marching Algorithm”. In: *Numerische Mathematik* 116.3, pp. 357–381.

- Benmansour, F. and L. D. Cohen (Feb. 2009a). “Fast Object Segmentation by Growing Minimal Paths from a Single Point on 2D or 3D Images”. In: *Journal of Mathematical Imaging and Vision* 33.2, pp. 209–221.
- (2009b). “Tubular Anisotropy Segmentation”. In: *Scale Space and Variational Methods in Computer Vision*. Ed. by X.-C. Tai, K. Mørken, M. Lysaker, and K.-A. Lie. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 14–25.
- (Apr. 2011). “Tubular Structure Segmentation Based on Minimal Path Method and Anisotropic Enhancement”. en. In: *International Journal of Computer Vision* 92.2, pp. 192–210.
- Bertrand, J. (2008). “Existence and uniqueness of optimal maps on Alexandrov spaces”. In: *Advances in Mathematics* 219.3, pp. 838–851.
- Biewald, L. (2020). *Experiment Tracking with Weights and Biases*. Software available from wandb.com.
- Bogachev, V. I. (2007). *Measure theory. Vol. I, II*. Berlin: Springer-Verlag, Vol. I: xviii+500 pp., Vol. II: xiv+575.
- Boyer, C., A. Chambolle, Y. D. Castro, V. Duval, F. de Gournay, and P. Weiss (Jan. 2019). “On Representer Theorems and Convex Regularization”. In: *SIAM Journal on Optimization* 29.2, pp. 1260–1281. arXiv: 1806.09810 [math.OC]³.
- Braides, A., G. Buttazzo, and I. Fragal (Aug. 2002). “Riemannian approximation of Finsler metrics”. In: *Asymptotic Analysis* 31.
- Bredies, K. and M. Carioni (Dec. 2019). “Sparsity of solutions for variational inverse problems with finite-dimensional data”. In: *Calculus of Variations and Partial Differential Equations* 59.1. arXiv: 1809.05045 [math.OC]⁴.
- Bredies, K., M. Carioni, S. Fanzon, and F. Romero (June 2021). “On the extremal points of the ball of the Benamou–Brenier energy”. In: *Bulletin of the London Mathematical Society*. arXiv: 1907.11589 [math.OC]⁵.
- (Mar. 2022). “A Generalized Conditional Gradient Method for Dynamic Inverse Problems with Optimal Transport Regularization”. In: *Foundations of Computational Mathematics*. arXiv: 2012.11706 [math.NA]⁶.
- Bredies, K. and S. Fanzon (Jan. 29, 2019). “An optimal transport approach for solving dynamic inverse problems in spaces of measures”. In: arXiv: 1901.10162 [math.FA]⁷.
- Bredies, K. and H. K. Pikkarainen (Mar. 2012). “Inverse problems in spaces of measures”. In: *ESAIM: Control, Optimisation and Calculus of Variations* 19.1, pp. 190–218.
- Brenier, Y. (1991). “Polar factorization and monotone rearrangement of vector-valued functions”. In: *Communications on pure and applied mathematics* 44.4, pp. 375–417.
- Bronstein, M. M., J. Bruna, T. Cohen, and P. Veličković (May 2021). *Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges*. en. arXiv:2104.13478 [cs, stat].

³<https://arxiv.org/abs/1806.09810>

⁴<https://arxiv.org/abs/1809.05045>

⁵<https://arxiv.org/abs/1907.11589>

⁶<https://arxiv.org/abs/2012.11706>

⁷<https://arxiv.org/abs/1901.10162>

- Burtscher, A. Y. (2015). *Length structures on manifolds with continuous Riemannian metrics*. arXiv: 1212.6962⁸.
- Calvo, D., M. Ortega, M. G. Penedo, and J. Rouco (July 2011). “Automatic detection and characterisation of retinal vessel tree bifurcations and crossovers in eye fundus images”. en. In: *Computer Methods and Programs in Biomedicine* 103.1, pp. 28–38.
- Castro, Y. de and F. Gamboa (Nov. 2012). “Exact reconstruction using Beurling minimal extrapolation”. In: *Journal of Mathematical Analysis and Applications* 395.1, pp. 336–354. arXiv: 1103.4951 [math.ST]⁹.
- Cavalletti, F. and M. Huesmann (2015). “Existence and uniqueness of optimal transport maps”. In: *Annales de l’Institut Henri Poincaré C, Analyse non linéaire* 32.6, pp. 1367–1377.
- Chambolle, A. and T. Pock (May 2011). “A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging”. en. In: *Journal of Mathematical Imaging and Vision* 40.1, pp. 120–145.
- (Mar. 2019). “Total roto-translational variation”. In: *Numerische Mathematik* 142.3, pp. 611–666. arXiv: 1709.09953 [math.NA]¹⁰.
- Chavignon, A. (Dec. 2021). “Microscopie par localisation ultrasonore dans les modèles d’accidents vasculaires cérébraux”. fr. PhD thesis. Université Paris sciences et lettres.
- Chavignon, A., H. Baptiste, V. Hingot, P. Lopez, T. Elliott, and O. Couture (Dec. 2020). *OPULM PALA*.
- Chen, D. and L. D. Cohen (Apr. 2016). “Vessel Tree Segmentation Via Front Propagation and Dynamic Anisotropic Riemannian Metric”. In: *ISBI 2016*. prague, Czech Republic.
- Chen, D., J.-M. Mirebeau, and L. D. Cohen (2016). “Vessel tree extraction using radius-lifted keypoints searching scheme and anisotropic fast marching method”. In: *Journal of Algorithms & Computational Technology* 10.4, pp. 224–234. eprint: <https://doi.org/10.1177/1748301816656289>.
- Chen, D., J.-M. Mirebeau, M. Shu, and L. D. Cohen (Aug. 2023). “Computing geodesic paths encoding a curvature prior for curvilinear structure tracking”. In: *Proceedings of the National Academy of Sciences* 120.33. Publisher: Proceedings of the National Academy of Sciences, e2218869120.
- Chen, R. T. Q. (2018). *torchdiffeq*.
- Chen, X., M. R. Lowerison, Z. Dong, N. V. Chandra Sekaran, D. A. Llano, and P. Song (Aug. 2023). “Localization Free Super-Resolution Microbubble Velocimetry Using a Long Short-Term Memory Neural Network”. en. In: *IEEE Transactions on Medical Imaging* 42.8, pp. 2374–2385.
- Chen, X., M. R. Lowerison, Z. Dong, A. Han, and P. Song (Apr. 2022). “Deep Learning-Based Microbubble Localization for Ultrasound Localization Microscopy”. In: *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* 69.4. Conference Name: IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, pp. 1312–1325.

⁸<https://arxiv.org/abs/1212.6962>

⁹<https://arxiv.org/abs/1103.4951>

¹⁰<https://arxiv.org/abs/1709.09953>

- Cohen, E., L. Cohen, T. Deffieux, and M. Tanter (Jan. 2018). “An Isotropic Minimal Path Based Framework for Segmentation and Quantification of Vascular Networks”. In: pp. 499–513.
- Cohen, L. (2001). “Multiple contour finding and perceptual grouping using minimal paths”. en. In: *Proceedings IEEE Workshop on Variational and Level Set Methods in Computer Vision*. Vancouver, BC, Canada: IEEE Computer Soc, pp. 89–96.
- Cohen, L. D. and T. Deschamps (2007). “Segmentation of 3D tubular objects with adaptive front propagation and minimal tree extraction for 3D medical imaging”. In: *Computer Methods in Biomechanics and Biomedical Engineering* 10.4. PMID: 17671862, pp. 289–305. eprint: <https://doi.org/10.1080/10255840701328239>.
- Cohen, L. D. and R. Kimmel (Aug. 1997). “Global Minimum for Active Contour Models: A Minimal Path Approach”. en. In: *International Journal of Computer Vision* 24.1, pp. 57–78.
- Cohen, T. and M. Welling (June 2016). “Group Equivariant Convolutional Networks”. en. In: *Proceedings of The 33rd International Conference on Machine Learning*. ISSN: 1938-7228. PMLR, pp. 2990–2999.
- Cohen, T. S., M. Geiger, J. Koehler, and M. Welling (Feb. 2018). *Spherical CNNs*. en. arXiv:1801.10130 [cs, stat].
- Cohen, T. S. and M. Welling (Dec. 2016). *Steerable CNNs*. en. arXiv:1612.08498 [cs, stat].
- Cormen, T. H., C. E. Leiserson, R. L. Rivest, and C. Stein (Sept. 2009). *Introduction to Algorithms, 3rd Edition*. English. 3rd edition. Cambridge, Massachusetts London, England: MIT Press.
- Couture, O., B. Besson, G. Montaldo, M. Fink, and M. Tanter (Oct. 2011). “Microbubble ultrasound super-localization imaging (MUSLI)”. en. In: *2011 IEEE International Ultrasonics Symposium*. Orlando, FL, USA: IEEE, pp. 1285–1287.
- Couture, O., V. Hingot, B. Heiles, P. Muleki-Seya, and M. Tanter (Aug. 2018). “Ultrasound Localization Microscopy and Super-Resolution: A State of the Art”. en. In: *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* 65.8, pp. 1304–1320.
- Crane, K., C. Weischedel, and M. Wardetzky (Oct. 2013). “Geodesics in heat: A new approach to computing distance based on heat flow”. In: *ACM Transactions on Graphics* 32.5, 152:1–152:11.
- Dencks, S. and G. Schmitz (Aug. 2023). “Ultrasound localization microscopy”. In: *Zeitschrift für Medizinische Physik*. Special Issue: Recent Advances in Ultrasound Imaging 33.3, pp. 292–308.
- Denoyelle, Q., V. Duval, and G. Peyré (Sept. 2016). “Support Recovery for Sparse Super-Resolution of Positive Measures”. In: *Journal of Fourier Analysis and Applications*.
- Deschamps, T. and L. D. Cohen (Dec. 2001). “Fast extraction of minimal paths in 3D images and applications to virtual endoscopy”. eng. In: *Medical Image Analysis* 5.4, pp. 281–299.

- Deschamps, T. and L. D. Cohen (2000). “Minimal Paths in 3D Images and Application to Virtual Endoscopy”. In: *Computer Vision — ECCV 2000*. Ed. by D. Vernon. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 543–557.
- Dosovitskiy, A. et al. (2021). “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *International Conference on Learning Representations*.
- Duits, R., M. Felsberg, G. Granlund, and B. ter Haar Romeny (Apr. 2007). “Image Analysis and Reconstruction using a Wavelet Transform Constructed from a Reducible Representation of the Euclidean Motion Group”. In: *International Journal of Computer Vision* 72, pp. 79–102.
- Duits, R., S. P. Meesters, J.-M. Mirebeau, and J. M. Portegies (2018). “Optimal paths for variants of the 2D and 3D Reeds–Shepp car with applications in image analysis”. In: *Journal of Mathematical Imaging and Vision* 60.6, pp. 816–848.
- Duval, V. and G. Peyré (Oct. 2014). “Exact Support Recovery for Sparse Spikes Deconvolution”. In: *Foundations of Computational Mathematics* 15.5, pp. 1315–1355. arXiv: 1306.6909 [math.OA]¹¹.
- Duval, V. and R. Tovey (2022). “Dynamical Programming for off-the-grid dynamic Inverse Problems”. In: arXiv: 2112.11378 [math.OA]¹².
- Ennaji, H., Y. Quéau, and A. Elmoataz (Mar. 2022). “A primal-dual algorithm for computing finler distances and applications”. working paper or preprint.
- Frangi, A. F., W. J. Niessen, K. L. Vincken, and M. A. Viergever (1998). “Multiscale vessel enhancement filtering”. In: *Medical Image Computing and Computer-Assisted Intervention — MICCAI’98*. Ed. by W. M. Wells, A. Colchester, and S. Delp. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 130–137.
- Frank, M. and P. Wolfe (Mar. 1956). “An algorithm for quadratic programming”. In: *Naval Research Logistics Quarterly* 3.1-2, pp. 95–110.
- Gigli, N. et al. (2012). “Optimal maps in non branching spaces with Ricci curvature bounded from below”. In: *Geom. Funct. Anal* 22.4, pp. 990–999.
- Hairer, E., S. Nørsett, and G. Wanner (2008). *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer Series in Computational Mathematics. Springer Berlin Heidelberg.
- Harpur, S. et al. (Oct. 2019). “Super-Resolution Ultrasound Image Filtering with Machine-Learning to Reduce the Localization Error”. en. In: *2019 IEEE International Ultrasonics Symposium (IUS)*. Glasgow, United Kingdom: IEEE, pp. 2118–2121.
- He, K., X. Zhang, S. Ren, and J. Sun (2015). *Deep Residual Learning for Image Recognition*.
- Heiles, B., A. Chavignon, V. Hingot, P. Lopez, E. Teston, and O. Couture (May 2022). “Performance benchmarking of microbubble-localization algorithms for ultrasound localization microscopy”. eng. In: *Nature Biomedical Engineering* 6.5, pp. 605–616.
- Heitz, M., N. Bonneel, D. Coeurjolly, M. Cuturi, and G. Peyré (Jan. 2021). “Ground Metric Learning on Graphs”. In: *Journal of Mathematical Imaging and Vision* 63.1, pp. 89–107.

¹¹<https://arxiv.org/abs/1306.6909>

¹²<https://arxiv.org/abs/2112.11378>

- Hervella, A., J. Rouco, J. Novo, M. Gonzalez, and M. Ortega (Nov. 2019). “Deep multi-instance heatmap regression for the detection of retinal vessel crossings and bifurcations in eye fundus images”. In: *Computer Methods and Programs in Biomedicine* 186, p. 105201.
- Jerman, T., F. Pernus, B. Likar, and Z. Spiclin (Sept. 2016). “Enhancement of Vascular Structures in 3D and 2D Angiographic Images”. en. In: *IEEE Transactions on Medical Imaging* 35.9, pp. 2107–2118.
- Laville, B., L. Blanc-Féraud, and G. Aubert (Dec. 2021). “Off-The-Grid Variational Sparse Spike Recovery: Methods and Algorithms”. en. In: *Journal of Imaging* 7.12, p. 266.
- (May 2023). “Off-the-Grid Curve Reconstruction through Divergence Regularization: An Extreme Point Result”. In: *SIAM Journal on Imaging Sciences* 16.2, pp. 867–885.
- Leconte, A. et al. (Aug. 2023). *A Tracking prior to Localization workflow for Ultrasound Localization Microscopy*. en. arXiv:2308.02724 [physics].
- LeCun, Y. et al. (Dec. 1989). “Backpropagation Applied to Handwritten Zip Code Recognition”. In: *Neural Computation* 1.4. Conference Name: Neural Computation, pp. 541–551.
- Li, H. and A. Yezzi (2007). “Vessels as 4-D curves: Global minimal 4-D paths to extract 3-D tubular surfaces and centerlines”. In: *IEEE transactions on medical imaging* 26.9, pp. 1213–1223.
- Malladi, R. and J. Sethian (1998). “A real-time algorithm for medical shape recovery”. In: *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pp. 304–310.
- Mirebeau, J.-M. (2019). “Riemannian Fast-Marching on Cartesian Grids, Using Voronoi’s First Reduction of Quadratic Forms”. In: *SIAM Journal on Numerical Analysis* 57.6, pp. 2608–2655. eprint: <https://doi.org/10.1137/17M1127466>.
- (Nov. 2023). *Mirebeau/MVA_ImagePDE*. original-date: 2021-01-18T08:26:34Z.
- Müllner, D. (2011). “Modern hierarchical, agglomerative clustering algorithms”. In: *CoRR* abs/1109.2378.
- Oliveira, A., S. Pereira, and C. A. Silva (Dec. 2018). “Retinal vessel segmentation based on Fully Convolutional Neural Networks”. In: *Expert Systems with Applications* 112, pp. 229–242.
- Parthasarathy, R. (July 2012). “Rapid, accurate particle tracking by calculation of radial symmetry centers”. en. In: *Nature Methods* 9.7. Number: 7 Publisher: Nature Publishing Group, pp. 724–726.
- Pedano, N. et al. (2016). *The Cancer Genome Atlas Low Grade Glioma Collection (TCGA-LGG) (Version 3)*.
- Peyré, G. and M. Cuturi (Mar. 2020). *Computational Optimal Transport*. en. arXiv:1803.00567 [stat].
- Peyré, G., M. Péchaud, R. Keriven, and L. D. Cohen (2010). “Geodesic Methods in Computer Vision and Graphics”. en. In: *Arxiv*.
- Pham, D. H., V. Pustovalov, and D. Kouamé (July 2023). “The Performance Improvement of Ultrasound Localization Microscopy (ULM) Using the Robust Principal Component Analysis (RPCA)”. en. In: *2023 45th Annual International*

- Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. Sydney, Australia: IEEE, pp. 1–4.
- Popovici, A. M. and J. A. Sethian (Mar. 2002). “3-D imaging using higher order fast marching traveltimes”. In: *GEOPHYSICS* 67.2. Publisher: Society of Exploration Geophysicists, pp. 604–609.
- Potter, S. F. and M. K. Cameron (Jan. 2021). “Jet Marching Methods for Solving the Eikonal Equation”. In: *SIAM Journal on Scientific Computing* 43.6. Publisher: Society for Industrial and Applied Mathematics, A4121–A4146.
- Pratt, H. et al. (Dec. 2017). “Automatic Detection and Distinction of Retinal Vessel Bifurcations and Crossings in Colour Fundus Photography”. en. In: *Journal of Imaging* 4.1, p. 4.
- Reeds, J. A. and L. A. Shepp (1990). “Optimal paths for a car that goes both forwards and backwards.” In: *Pacific Journal of Mathematics* 145.2, pp. 367–393.
- Rentmeesters, Q. (Dec. 2011). “A gradient method for geodesic data fitting on some symmetric Riemannian manifolds”. In: *2011 50th IEEE Conference on Decision and Control and European Control Conference*. ISSN: 0743-1546, pp. 7141–7146.
- Rickett, J. and S. Fomel (1999). “A second-order fast marching eikonal solver”. In: *Stanford Exploration Project Report* 100, pp. 287–293.
- Ronneberger, O., P. Fischer, and T. Brox (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*.
- Santambrogio, F. (2015). “Optimal transport for applied mathematicians”. In: *Birkäuser, NY* 55.58-63, p. 94.
- Scarvelis, C. and J. Solomon (2022). *Riemannian Metric Learning via Optimal Transport*.
- Schmitzer, B., K. P. Schäfers, and B. Wirth (Feb. 20, 2019). “Dynamic Cell Imaging in PET with Optimal Transport Regularization”. In: arXiv: 1902.07521 [cs.CV]¹³.
- Sethian, J. A. (1996). “A fast marching level set method for monotonically advancing fronts.” In: *Proceedings of the National Academy of Sciences* 93.4, pp. 1591–1595. eprint: <https://www.pnas.org/doi/pdf/10.1073/pnas.93.4.1591>.
- Siepmann, M., G. Schmitz, J. Bzyl, M. Palmowski, and F. Kiessling (Oct. 2011). “Imaging tumor vascularity by tracing single microbubbles”. In: *2011 IEEE International Ultrasonics Symposium*. ISSN: 1948-5727, pp. 1906–1909.
- Skalski, P. (2019). *Make Sense*. <https://github.com/SkalskiP/make-sense/>.
- Smets, B., J. Portegies, E. Bekkers, and R. Duits (Jan. 2023). “PDE-based Group Equivariant Convolutional Neural Networks”. en. In: *Journal of Mathematical Imaging and Vision* 65.1. arXiv:2001.09046 [cs, math, stat], pp. 209–239.
- Song, P. et al. (Feb. 2018). “Improved Super-Resolution Ultrasound Microvessel Imaging with Spatiotemporal Nonlocal Means Filtering and Bipartite Graph-Based Microbubble Tracking”. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 65.2, pp. 149–167.
- Tetteh, G. et al. (Dec. 2020). “DeepVesselNet: Vessel Segmentation, Centerline Prediction, and Bifurcation Detection in 3-D Angiographic Volumes”. en. In: *Frontiers in Neuroscience* 14, p. 592352.

¹³<https://arxiv.org/abs/1902.07521>

- Varadhan, S. R. S. (Nov. 1967a). “Diffusion processes in a small time interval”. en. In: *Communications on Pure and Applied Mathematics* 20.4, pp. 659–685.
- (1967b). “On the behavior of the fundamental solution of the heat equation with variable coefficients”. en. In: *Communications on Pure and Applied Mathematics* 20.2. [_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.3160200210](https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.3160200210), pp. 431–455.
- Villani, C. (2021). *Topics in Optimal Transportation*. Graduate Studies in Mathematics. American Mathematical Society.
- Villani, C. (2009). *Optimal transport: old and new*. Vol. 338. Springer.
- Wang, G. et al. (Mar. 2023). “Automatic vessel crossing and bifurcation detection based on multi-attention network vessel segmentation and directed graph search”. en. In: *Computers in Biology and Medicine* 155, p. 106647.
- Yang, F., L. Chai, D. Chen, and L. D. Cohen (Dec. 2018). “Geodesics via asymmetric Heat diffusion based on Finsler Metric”. In: *14th Asian Conference on Computer Vision (ACCV18)*. Proc. 14th Asian Conference on Computer Vision (ACCV18). Perth, Australia.
- Yang, F. and L. D. Cohen (June 2016). “Geodesic Distance and Curves Through Isotropic and Anisotropic Heat Equations on Images and Surfaces”. In: *Journal of Mathematical Imaging and Vision*. Special Issue on Shape Analysis Beyond the Eikonal Equation 55.2, pp. 210–228.
- Youn, J., M. L. Ommen, M. Bo Stuart, E. V. Thomsen, N. Bent Larsen, and J. A. Jensen (Oct. 2019). “Ultrasound Multiple Point Target Detection and Localization using Deep Learning”. en. In: *2019 IEEE International Ultrasonics Symposium (IUS)*. Glasgow, United Kingdom: IEEE, pp. 1937–1940.
- Youn, J., M. L. Ommen, M. B. Stuart, E. V. Thomsen, N. B. Larsen, and J. A. Jensen (Dec. 2020). “Detection and Localization of Ultrasound Scatterers Using Convolutional Neural Networks”. en. In: *IEEE Transactions on Medical Imaging* 39.12, pp. 3855–3867.
- Zhang, J., B. Dashtbozorg, E. Bekkers, J. P. W. Pluim, R. Duits, and B. M. ter Haar Romeny (2016). “Robust Retinal Vessel Segmentation via Locally Adaptive Derivative Frames in Orientation Scores”. In: *IEEE Transactions on Medical Imaging* 35.12, pp. 2631–2644.
- Zhang, W., K. Itoh, J. Tanida, and Y. Ichioka (Nov. 1990). “Parallel distributed processing model with local space-invariant interconnections and its optical architecture”. EN. In: *Applied Optics* 29.32. Publisher: Optica Publishing Group, pp. 4790–4797.
- Zhao, H. (May 2004). “A fast sweeping method for Eikonal equations”. en. In: *Mathematics of Computation* 74.250, pp. 603–627.
- (Jan. 2006). “Parallel implementation of the fast sweeping method”. In: *International Journal of Computer Mathematics - IJCM* 25.
- Zhou, S. et al. (May 2021). “A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies with Progress Highlights, and Future Promises”. English (US). In: *Proceedings of the Institute of Radio Engineers* 109.5, pp. 820–838.

Geodesic and Learning methods for Ultrasound Localization Microscopy

Résumé

La méthode de Microscopie par Localisation Ultrasonore (ULM) est une nouvelle méthode d'imagerie médicale super-résolue qui permet d'outrepasser le compromis précision/distance de pénétration dans les tissus pour l'imagerie du réseau vasculaire. Ce nouveau type d'images pose de nouveaux enjeux mathématiques, notamment pour la segmentation et l'analyse de ses images, étapes nécessaires pour arriver à des méthodes pour le diagnostic. Notre travail se positionne à l'intersection des méthodes géodésiques et des méthodes de Machine Learning. Dans cette thèse nous apportons trois contributions. Une première de ces contributions est centrée autour des contraintes liées aux images ULM et propose le tracking de l'entièreté de l'arbre vasculaire en passant par la détection des point-clés des vaisseaux sanguins apparaissant sur l'image. La deuxième contribution de cette thèse porte sur l'apprentissage de la définition de métriques Riemanniennes pour traiter des tâches de segmentation sur des données d'IRM cérébraux et d'images du fond de l'oeil. La dernière partie de notre travail porte sur un problème inverse pour la reconstruction de trajectoires d'agents de contraste dans des images médicales dans le contexte de la super-résolution sans-grille.

Mots clés : Computer Vision, Geodesic distances, Geodesic curves, Inverse Problems, Machine learning

Geodesic and Learning methods for Ultrasound Localization Microscopy

Résumé

Ultrasound Localization Microscopy is a new method in super-resolved Medical Imaging that allow us to overcome compromise between precision and penetration distance in the tissues for the imaging of the vascular network. This new type of images raises new mathematical questions, especially for the segmentaton and analysis, necessary steps to achieve medical diagnostic of patients. Our work is positioned at the intersection of geodesic and Machine Learning methods. In this thesis, we make three contributions. The first of these is centered on the constraints linked to ULM images and proposes the tracking of the entire vascular tree through the detection of key points of blood vessels appearing on the image. The second contribution of this thesis deals with learning to define Riemannian metrics to handle segmentation tasks on brain MRI data and eye fundus images. The final part of our work focuses on an inverse problem for reconstructing contrast agent trajectories in medical images in the context of grid-free super-resolution.

Mots clés : Computer Vision, Geodesic distances, Geodesic curves, Inverse Problems, Machine learning

RÉSUMÉ

La méthode de Microscopie par Localisation Ultrasonore (ULM) est une nouvelle méthode d'imagerie médicale super-résolue qui permet d'outrepasser le compromis précision/distance de pénétration dans les tissus pour l'imagerie du réseau vasculaire. Ce nouveau type d'images pose de nouveaux enjeux mathématiques, notamment pour la segmentation et l'analyse de ses images, étapes nécessaires pour arriver à des méthodes pour le diagnostic. Notre travail se positionne à l'intersection des méthodes géodésiques et des méthodes de Machine Learning. Dans cette thèse nous apportons trois contributions. Une première de ces contributions est centrée autour des contraintes liées aux images ULM et propose le tracking de l'entièreté de l'arbre vasculaire en passant par la détection des point-clés des vaisseaux sanguins apparaissant sur l'image. La deuxième contribution de cette thèse porte sur l'apprentissage de la définition de métriques Riemanniennes pour traiter des tâches de segmentation sur des données d'IRM cérébraux et d'images du fond de l'oeil. La dernière partie de notre travail porte sur un problème inverse pour la reconstruction de trajectoires d'agents de contraste dans des images médicales dans le contexte de la super-résolution sans-grille.

MOTS CLÉS

Vision par ordinateur, Distances géodésiques, Courbes géodésiques, Problèmes inverses, Machine learning

ABSTRACT

Ultrasound Localization Microscopy is a new method in super-resolved Medical Imaging that allow us to overcome compromise between precision and penetration distance in the tissues for the imaging of the vascular network. This new type of images raises new mathematical questions, especially for the segmentaton and analysis, necessary steps to achieve medical diagnostic of patients. Our work is positioned at the intersection of geodesic and Machine Learning methods. In this thesis, we make three contributions. The first of these is centered on the constraints linked to ULM images and proposes the tracking of the entire vascular tree through the detection of key points of blood vessels appearing on the image. The second contribution of this thesis deals with learning to define Riemannian metrics to handle segmentation tasks on brain MRI data and eye fundus images. The final part of our work focuses on an inverse problem for reconstructing contrast agent trajectories in medical images in the context of grid-free super-resolution.

KEYWORDS

Computer Vision, Geodesic distances, Geodesic curves, Inverse Problems, Machine learning