



HAL
open science

Techniques de mise en correspondance et détection de changements

Laurent Garcin

► **To cite this version:**

Laurent Garcin. Techniques de mise en correspondance et détection de changements. Mathématiques [math]. École normale supérieure de Cachan - ENS Cachan, 2005. Français. NNT : . tel-00133078

HAL Id: tel-00133078

<https://theses.hal.science/tel-00133078>

Submitted on 23 Feb 2007

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.

Techniques de Mise en Correspondance
et
Détection de Changements

Laurent GARCIN
Directeur de thèse : Laurent Younes

23 juin 2005

Table des matières

1	Introduction	7
1.1	Modélisation des déformations	7
1.1.1	Mise en correspondance de surfaces	8
1.1.2	Mise en correspondance de primitives	8
1.2	Plan de la thèse	9
2	Théorie des grandes déformations	11
2.1	Problématique	11
2.1.1	Action de groupe	11
2.1.2	Distance sur l'espace des objets	12
2.1.3	Notations	12
2.2	Distances géodésiques	13
2.2.1	Energie d'un chemin	13
2.2.2	Condition d'invariance	13
2.2.3	Vitesse et chemin effectif	15
2.3	Les métamorphoses	16
2.3.1	Définition et action de groupe	16
2.3.2	Actions infinitésimales	17
2.3.3	Construction d'une métrique riemannienne	18
2.3.4	Comparaison avec l'approche précédente	19
2.4	Exemples	19
2.4.1	Mise en correspondance d'images	19
2.4.2	Mise en correspondance de points	22
2.5	Existence d'un minimum de l'énergie	24
3	Mise en correspondance d'images et de modèles numériques d'élévation	27
3.1	Déformations dans le cadre de la détection de changement	27
3.1.1	Choix de l'énergie et des inconnues	27
3.1.2	Régularisation par un noyau gaussien	28
3.1.3	Décomposition de la déformation dans une base appropriée	30
3.2	Algorithme	38
3.2.1	Version discrétisée de l'énergie	38
3.2.2	Calcul du gradient en la déformation	38
3.2.3	Calcul du gradient en l'image	40
3.3	Conclusion	41

4	Mise en correspondance de primitives	55
4.1	Modélisation de primitives	55
4.2	Interpolation par splines	56
4.2.1	Notations	56
4.2.2	Problème générique	56
4.2.3	Interpolation par spline au premier ordre	57
4.2.4	Algorithme de moindres carrés	58
4.2.5	Prise en compte d'une transformation affine	58
4.3	Perspectives	59
4.4	Conclusion	59
5	Appariement et déformation	61
5.1	Formalisation du problème et définition d'une énergie de mise en correspondance	61
5.1.1	Modélisation hiérarchique des ensembles	61
5.1.2	Modélisation des appariements	62
5.1.3	Modélisation de la déformation	64
5.1.4	Définition d'une énergie de mise en correspondance	65
5.2	Minimisation de l'énergie	65
5.2.1	Mise à jour des correspondances	65
5.2.2	Mise à jour de la déformation	66
5.2.3	Algorithme générique	66
5.3	Etude d'un cas simple : appariement de points en multi-vues	68
5.4	Etude d'un cas plus complexe	71
5.4.1	Définition des objets	71
5.4.2	Matrices d'appariement	74
5.4.3	Espace des déformations	74
5.4.4	Energie de mise en correspondance	75
5.4.5	Etapas d'appariement	75
5.4.6	Etape de déformation	76
5.4.7	Algorithme de minimisation	77
5.5	Applications	78
5.5.1	Mise en correspondance de segments	78
5.5.2	Mise en correspondance de lignes	81
5.5.3	Mise en correspondance de polygones	81
5.5.4	Mise en correspondance de facettes	82
5.6	Conclusion	85
6	Mise en correspondance géodésique de primitives	89
6.1	Mise en correspondance géodésique de vecteurs	89
6.2	Mise en correspondance géodésique de vecteurs unitaires	91
6.3	Formalisation par les splines	93
6.4	Algorithme de minimisation de l'énergie	94
6.4.1	Minimisation sur le champ de déformation	94
6.4.2	Minimisation sur les trajectoires	95
6.4.3	Schéma algorithmique	98
6.5	Adaptation de la méthode à la mise en correspondance de vecteurs unitaires	99
6.5.1	Forme du champ de vitesse	99
6.5.2	Minimisation de l'énergie en la déformation	99

6.5.3	Minimisation de l'énergie en les trajectoires	100
6.6	Applications	100
6.7	Perspectives	100
7	Mise en correspondance géodésique de primitives par quantification	105
7.1	Problématique	105
7.2	Algorithme de quantification	105
7.2.1	K-means	106
7.2.2	Soft K-means	106
7.3	Mise en correspondance de primitives	108
7.3.1	Energie d'adéquation avec les quanta	108
7.3.2	Energie liée au difféomorphisme	109
7.4	Minimisation de l'énergie	109
7.4.1	Etape de quantification	109
7.4.2	Calcul de la déformation	109
7.4.3	Calcul des trajectoires	110
7.4.4	Algorithme	110
7.5	Résultats	111
7.5.1	Consistance	112
7.5.2	Robustesse	112
7.6	Perspectives	112
7.7	Conclusion	115
8	Conclusion	121
8.1	Mise en correspondance de surfaces	121
8.2	Mise en correspondance de primitives	121
A	Construction des MNE	123
A.1	Méthode de corrélation simple	123
A.2	Méthode par maxima locaux de corrélation	124
B	Espaces de Hilbert à noyau reproduisant	127
B.1	De L^2 aux RKHS	127
B.2	Cas usuel	128
B.2.1	Construction d'un RKHS à partir du théorème de Mercer	128
B.2.2	Construction d'un RKHS à partir d'un noyau invariant par translation	129
B.3	Extension au premier ordre	130
B.3.1	Construction d'un RKHS de premier ordre à partir du théorème de Mercer	130
B.3.2	Construction d'un RKHS de premier ordre à partir d'un noyau invariant par translation	132

Résumé

L'objet de cette thèse est la détection de changements dans différents types de données. La notion de "changement" est un peu vague. Par exemple, deux images de la même scène peuvent être très différentes suivant l'angle de prise de vue et pourtant elles représentent bel et bien la même scène. De même, deux modélisations 3D d'une même scène peuvent différer grandement en fonction du degré de précision ou de généralisation avec lequel elles ont été construites. La notion de changement dépend aussi du type de données à comparer. Comparer une base de données 3D définies par des primitives (points, segments, ...) avec un modèle numérique d'élévation (MNE) n'est pas la même chose que comparer deux prises de vue d'une scène.

Dans le cadre de cette thèse, on se limitera donc à comparer des données du même type, à savoir :

- des surfaces (i.e. des images ou des MNE), du type $z = f(x, y)$
- des primitives 2D ou 3D (i.e. des points, des segments ou d'autres primitives plus complexes).

Les méthodes adoptées dans chacun de ces deux cas seront bien entendu différentes de par la nature des données mais ce seront toujours des techniques de mise en correspondance. C'est-à-dire que l'on souhaitera mettre en correspondance les surfaces ou les primitives. Dans le cadre de la mise en correspondance de surfaces, on utilisera une mise en correspondance dense, mettant en correspondance chaque point des deux surfaces. Dans le cadre de la mise en correspondance de primitives, on aura à estimer à la fois un appariement et une déformation permettant de passer de l'ensemble de primitives initial à l'ensemble de primitives final.

La mise en correspondance dense nous fournira un champ de déformation permettant de passer des données à une date aux données à une date ultérieure. Il faudra donc une étape d'analyse du champ de déformation afin de détecter et caractériser les changements.

Dans le cadre de la mise en correspondance de primitives, une matrice de confusion nous donnera directement accès aux changements (disparition ou apparition de primitives) ou bien il faudra à nouveau analyser la déformation pour retrouver les changements.

Chapitre 1

Introduction

1.1 Modélisation des déformations

Que ce soit pour apparier des images et des primitives, on sera amené à modéliser des déformations permettant de passer d'un objet à l'autre. On emploiera dans notre travail des méthodes de déformations non-rigides comportant éventuellement une partie rigide (i.e. affine). Pour prendre en compte des déformations souples, on modélisera ces déformations :

- soit de manière dense dans le cadre de la mise en correspondance de surfaces, c'est-à-dire en estimant un champ de déformation,
- soit par interpolation à bases de splines [11, 57] dans le cadre de la mise en correspondance de primitives notamment en utilisant la théorie des noyaux reproduisants.

Traditionnellement, on modélise les déformations par ce qu'on appellera par la suite des *petites* déformations, c'est-à-dire des applications qui à un point x associent un nouveau point $u(x)$ en imposant sur u certaines conditions de régularité par exemple. Ces applications u peuvent être par exemple des transformations affines, provenir d'un champ de déplacement ($u(x) = x + v(x)$), ou bien encore être générés par une interpolation par splines ($u(x) = x + \sum_i \alpha_i S(x_i, x)$).

Néanmoins, ces méthodes ne permettent pas de générer de grandes déformations et conduisent parfois à des “catastrophes” géométriques car on ne peut généralement pas imposer de conditions topologiques sur ce type de déformations, tout au plus une certaine régularité. On aura donc recours à une théorie des *grandes* déformations introduite dans [35, 43, 36, 25] connue aussi sous l'appellation de “computational anatomy” pour ses liens avec l'imagerie médicale. On pourra se convaincre de l'utilité de cette théorie notamment en lisant [43, 42] qui compare petites et grandes déformations. Cette théorie modélise les déformations par des flots de difféomorphismes ou bien, pour faire le lien avec les *petites* déformations, comme une somme infinitésimale de déplacements du . Ces deux approches sont équivalentes car le flot de la déformation Φ et la vitesse instantanée $v = du/dt$ ¹ de la déformation sont reliés par l'équation de transport

$$\frac{d\Phi}{dt}(t, x) = v(t, \Phi(t, x))$$

¹Il faudra bien sûr donner un sens à cette expression.

On peut trouver une approche différente et plus générale faisant intervenir la notion d'action de groupe de déformations sur des objets (images, landmarks, ...) dans [42, 60, 53, 54]. Cette approche se trouve dans la lignée des travaux sur les “modèles déformables” (deformable templates) initiés par Grenander [31]. La différence est que l'on considère une somme d'actions infinitésimales des éléments du groupe de déformations sur les objets. Cette technique permet de définir une structure riemannienne sur l'ensemble auquel appartiennent les objets à apparier par l'intermédiaire de chemins de distance minimale sur cet ensemble, ce qui justifiera la dénomination de “mise en correspondance géodésique” qu'on emploiera par la suite. Un des avantages de ces techniques par rapport aux techniques de ‘*petites* déformations – outre le fait que l'on peut retrouver des déformations à plus grande échelle – est qu'on est assurée que la déformation est un difféomorphisme, et donc une bijection, ce qui évite tout problème de recouvrement indésirable dans certaines applications.

1.1.1 Mise en correspondance de surfaces

Notre but sera d'apparier des surfaces (images, MNE, ...) mais on sera en particulier intéressé par la caractérisation des déformations permettant de passer d'une image à l'autre. Cela élimine d'avance toutes les méthodes ne permettant pas de calculer de telles déformations, notamment toutes celles faisant appel à une modélisation probabiliste des images [30, 32].

Le calcul de la déformation pourrait se faire par des techniques de flot optique (cf. [6] pour une étude exhaustive des techniques de calcul de flot optique) mais elles sont peu adéquates pour retrouver de grandes déformations (notamment des prises de vue d'une même scène espacées dans le temps).

On utilisera donc des techniques de grandes déformations en faisant agir un groupe de difféomorphismes sur le paramétrage de la surface. On montrera d'ailleurs que l'équivalent en petites déformations de cette méthode est justement le calcul du flot optique qui lui est donc inférieur.

L'approche multi-résolution ayant par ailleurs donné de bons résultats en ce qui concerne les techniques de flot optique [3, 10], on adaptera donc cette approche à notre cadre théorique, notamment par l'analyse de la déformation sur des bases d'ondelettes.

1.1.2 Mise en correspondance de primitives

Il peut être coûteux de comparer des surfaces par le biais d'une mise en correspondance dense. Quand on compare deux scènes, il peut être plus judicieux de comparer seulement les éléments caractéristiques de cette scène. Dans notre cas, il s'agira de primitives extraites de données 2D ou 3D. On cherchera à nouveau à estimer une déformation entre les deux ensembles de primitives à comparer mais un nouveau problème se pose qui est l'appariement des primitives de chaque ensemble deux à deux. Il est aussi essentiel de pouvoir gérer les “outliers” (objets manquants ou surnuméraires). On veut donc être capable de trouver une matrice de confusion représentant les appariements entre les objets des deux ensembles. Le problème peut donc être séparé en deux parties :

- l'appariement des primitives en tenant compte des outliers si possible,
- l'estimation de la déformation optimale entre les deux ensembles de primitives connaissant les correspondances entre les primitives deux à deux.

Dans le cas où l'on connaît déjà par avance l'appariement entre primitives, on parlera de mise en correspondance non ambiguë, et de mise en correspondance ambiguë dans le cas contraire.

En ce qui concerne la mise en correspondance de “landmarks”, le cadre des grandes déformations a été combiné à des techniques à base de splines pour donner des splines géodésiques [61]. On en proposera une extension pour la mise en correspondance de vecteurs.

Cependant, la mise en correspondance géodésique ne permet pas d'apparier les primitives deux à deux dans l'ensemble de départ et dans l'ensemble d'arrivée en même temps que l'estimation de la déformation, c'est-à-dire, pour reprendre la terminologie indiquée plus haut, qu'il ne fonctionne que dans un cadre non ambigu. L'étape d'appariement doit être effectuée avant la mise en correspondance mais on perd alors la robustesse qui résulte de l'estimation conjointe de l'appariement et de la déformation. On sera donc amenés à explorer des techniques de *petites* déformations qui permettent l'estimation conjointe de l'appariement et de la déformation [17, 46, 28, 19].

On verra néanmoins comment s'affranchir de la contrainte de l'appariement grâce à la quantification des primitives à apparier [18, 20, 21] mais cela nécessitera quand même une étude *a posteriori* de la déformation.

1.2 Plan de la thèse

Dans un premier temps, on introduira les bases de la “mise en correspondance géodésique” (chap. 2) qui permet de définir des métriques sur des ensembles d'objets variés et par conséquent de calculer une distance entre deux objets, ce qui permet de les comparer. On fera le lien avec la théorie des métamorphoses [55]. On proposera en particulier deux applications pour la mise en correspondance d'images et de points.

On étudiera ensuite plus attentivement l'application qui peut en être faite pour la mise en correspondance de MNE ou d'images et on discutera en particulier des normes de régularisation qui interviennent dans la mise en correspondance (chap. 3) avec une possibilité de les définir par apprentissage.

On passera ensuite à la mise en correspondance non ambiguë de primitives – c'est-à-dire sans appariement – (chap. 4) en décrivant notamment la mise en correspondance de caractéristiques vectorielles des objets à mettre en correspondance (tangentes, normales) grâce à des techniques d'interpolation par splines.

On introduira ensuite un modèle d'objet nous permettant d'apparier des primitives complexes et une technique basée sur le “soft assign” [28] permettant de trouver les correspondances entre objets (chap. 5).

On reviendra ensuite à la mise en correspondance géodésique (chap. 6) mais dans le cadre de la mise en correspondance de caractéristiques ponctuelles ou vectorielles.

Enfin, on verra une méthode de mise en correspondance de primitives permettant de s'affranchir du problème de l'appariement grâce à la quantification des primitives (chap. 7).

L'annexe A expose la méthode employée pour construire les MNE à apparier grâce à des techniques de corrélations en multi-vues (plusieurs prises de vue d'une même scène 3D).

L'annexe B donne un aperçu des espaces de Hilbert à noyau reproduisant ainsi qu'une extension pour tenir compte de l'aspect différentiel des éléments du Hilbert. On appellera ces espaces des espaces de Hilbert à noyau reproduisant de premier ordre.

Chapitre 2

Théorie des grandes déformations

Le but de ce chapitre est d'introduire la notion de distance géodésique sur un ensemble d'objets sur lesquels agissent un groupe. Concrètement, les objets seront des MNE ou des images (cf. chap. 3) ou des ensembles de landmarks (cf. chap. 6, 7) et le groupe sera un groupe de déformations. Le but sera alors de trouver la déformation minimisant une distance géodésique entre les deux objets, cette déformation représentant les changements d'un objet à l'autre. On peut retrouver le cadre général de cette théorie dans [42]. Plus précisément, on dira que deux objets I_0 et I_1 sont proches s'il existe une déformation g telle que $g.I_0 \approx I_1$ et $g \approx Id$.

2.1 Problématique

2.1.1 Action de groupe

Soit une variété \mathcal{I} considérée comme un espace d'objets sur laquelle agit un groupe G d'élément neutre e possédant une algèbre de Lie \mathfrak{G} . Dans la suite, on notera l'action d'un groupe G sur un ensemble d'objets \mathcal{I} de la manière suivante :

$$\begin{aligned} G \times \mathcal{I} &\rightarrow \mathcal{I} \\ (g, I) &\rightarrow g.I \end{aligned}$$

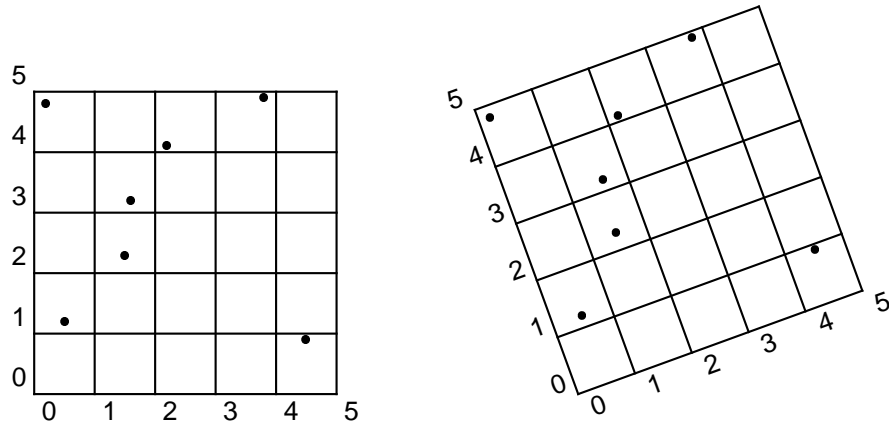
De plus G agit naturellement sur lui-même par l'intermédiaire de la loi de groupe.

Hypothèse 2.1. *On supposera que l'action de $G \times \mathcal{I} \rightarrow \mathcal{I}$ est C^1 .*

Le but est de comparer les objets de \mathcal{I} . On dira que deux éléments I_0 et I_1 de \mathcal{I} sont proches s'il existe $g \in G$ tel que $g \simeq e$ et $g.I_0 \simeq I_1$, autrement dit s'il existe un élément du groupe G suffisamment proche de l'élément neutre faisant correspondre les deux objets.

On introduit à cet effet l'ensemble $\mathcal{A} = G \times \mathcal{I}$ sur lequel G agit à nouveau de la manière suivante :

$$(g, I).h = (gh, h^{-1}.I)$$

FIG. 2.1 – Action d'une rotation sur un 7-uplet de \mathbb{R}^2

Remarque 2.2. On considère une action à droite du groupe G sur \mathcal{A} qui se traduit par le fait que $I.h = h^{-1}.I$.

Exemple 2.3. Un exemple simple pourrait être l'ensemble $(\mathbb{R}^d)^N$ des N -uplets de \mathbb{R}^d sur lequel agit le groupe des rotations de \mathbb{R}^d (cf. fig. 2.1). Dans ce cas, cela revient à faire un recalage rigide de deux ensembles de N points (l'appariement point à point étant supposé connu).

2.1.2 Distance sur l'espace des objets

On définit alors la distance $d(I, I')$ entre deux objets I et I' de \mathcal{I} comme la distance entre les orbites de (e, I) et (e, I') .

Définition 2.4. La distance d sur \mathcal{I} est telle que pour tout $I, I' \in \mathcal{I}$:

$$\begin{aligned} d(I, I') &= D((e, I).G, (e, I').G) \\ &= \inf\{D(a, a') : a = (g, g^{-1}.I), a' = (g', g'^{-1}.I'), g, g' \in G\} \end{aligned} \quad (2.1)$$

Il faut noter que, dans l'expression (2.1), la distance D reste à définir. Malheureusement d n'est pas dans le cas général une distance au sens propre du terme, à moins que l'on ne suppose que D vérifie une condition d'invariance par rapport à l'action du groupe G i.e. quelques soient $h, g, g' \in G$ et $I, I' \in \mathcal{I}$:

$$D((g, I).h, (g', I').h) = D((g, I), (g', I'))$$

2.1.3 Notations

A partir des actions de groupes de G sur \mathcal{I} et sur lui-même, on définit plusieurs applications qui nous seront utiles par la suite.

Définition 2.5. Pour $I \in \mathcal{I}$ et $g \in G$, on note $R_I^{\mathcal{I}}$ l'application :

$$\begin{aligned} R_I^{\mathcal{I}} : G &\rightarrow \mathcal{I} \\ g &\rightarrow g.I \end{aligned}$$

R_g^G l'application :

$$\begin{aligned} R_g^G : G &\rightarrow G \\ h &\rightarrow h.g = hg \end{aligned}$$

et R_g^I l'application :

$$\begin{aligned} R_g^I : I &\rightarrow I \\ I &\rightarrow g^{-1}.I \end{aligned}$$

2.2 Distances géodésiques

2.2.1 Energie d'un chemin

Soit \mathbf{a} un chemin différentiable sur \mathcal{A} , c'est-à-dire une application continue de $[0; 1]$ dans \mathcal{A} pour laquelle la dérivée temporelle $\frac{d\mathbf{a}}{dt}$ est définie pour tout $t \in [0; 1]$. Pour que cette dérivée temporelle soit bien définie, on suppose par la suite que \mathcal{A} est un sous-ensemble d'un espace de Banach.

Définition 2.6. Soit \mathcal{M} une variété. Pour $m \in \mathcal{M}$, on note $T_m\mathcal{M}$ l'espace tangent à \mathcal{M} en m , autrement dit l'ensemble contenant les dérivées $\frac{d\mathbf{m}}{dt}(t_0)$ pour tout $t_0 \in [0; 1]$ et tout chemin différentiable \mathbf{m} tel que $\mathbf{m}(t_0) = m$.

Ainsi on pourra parler de l'espace $T_a\mathcal{A}$ tangent à \mathcal{A} en a . De plus, si $a = (g, I)$, on a $T_a\mathcal{A} = T_gG \times T_I I$.

On définit l'énergie associée à un chemin sous la forme d'une somme d'énergies infinitésimales. Ces énergies infinitésimales sont associées à une famille de normes $(\|\cdot\|_a, a \in \mathcal{A})$ telle que, quelque soit $a \in \mathcal{A}$, $\|\cdot\|_a$ est une norme sur $T_a\mathcal{A}$.

Définition 2.7. L'énergie associée au chemin \mathbf{a} est donc la somme de ces énergies infinitésimales ou instantanées :

$$E(\mathbf{a}) = \int_0^1 \left\| \frac{d\mathbf{a}}{dt} \right\|_{\mathbf{a}(t)}^2 dt$$

A partir de cette énergie on peut définir le chemin géodésique entre deux points comme le chemin d'énergie minimale et l'on en tire la notion de distance géodésique.

Définition 2.8. La distance géodésique associée à l'énergie E est par définition :

$$D(a, a') = \inf \{ \sqrt{E(\mathbf{a})} : \mathbf{a}(0) = a, \mathbf{a}(1) = a' \}$$

2.2.2 Condition d'invariance

On a vu dans 2.1 que la "distance" d définie en (2.1) est réellement une distance que si on peut assurer que D est invariante par rapport à l'action de groupe. L'invariance de D peut être assurée par certaines conditions sur les normes $\|\cdot\|_a$.

On définit pour cela l'application de translation par g .

Définition 2.9. Pour $g \in G$, on note R_g l'application :

$$\begin{aligned} R_g : \mathcal{A} &\rightarrow \mathcal{A} \\ a &\rightarrow a.g \end{aligned}$$

Comme les actions de G sur \mathcal{I} et sur lui-même sont C^1 , l'application R_g est différentiable pour tout $g \in G$. Donc pour tout $g \in G$ et tout chemin différentiable \mathbf{a} sur \mathcal{A} , le translaté de \mathbf{a} par g i.e. $\mathbf{a}.g : t \rightarrow a(t).g$ est aussi différentiable. Pour tout $g \in G$, la différentielle de l'application R_g , à savoir $d_a R_g : T_a \mathcal{A} \rightarrow T_{a.g} \mathcal{A}$ vérifie alors la condition suivante pour tout chemin différentiable \mathbf{a} tel que $\mathbf{a}(t_0) = a$:

$$\frac{d\mathbf{a}.g}{dt}(t_0) = d_a R_g \cdot \frac{d\mathbf{a}}{dt}(t_0) \quad (2.2)$$

Proposition 2.10. La condition d'invariance de D par rapport à l'action de groupe est vérifiée si pour tout $a \in \mathcal{A}$, pour tout $t_0 \in [0; 1]$ et tout chemin différentiable \mathbf{a} tel que $\mathbf{a}(t_0) = a$

$$\left\| \frac{d\mathbf{a}}{dt}(t_0) \right\|_a = \left\| d_a R_g \cdot \frac{d\mathbf{a}}{dt}(t_0) \right\|_{a.g} \quad (2.3)$$

Démonstration. Si la condition d'invariance (2.3) est vérifiée, alors on peut écrire pour g dans G et tout chemin \mathbf{a} sur \mathcal{A} tel que $\mathbf{a}(0) = a$ et $\mathbf{a}(1) = a'$:

$$E(g.\mathbf{a}) = \int_0^1 \left\| \frac{d\mathbf{a}.g}{dt} \right\|_{\mathbf{a}(t).g}^2 dt = \int_0^1 \left\| \frac{d\mathbf{a}}{dt} \right\|_{\mathbf{a}(t)}^2 dt = E(\mathbf{a}) \quad (2.4)$$

et, par conséquent $D(a.g, a'.g) = D(a, a')$. \square

Proposition 2.11. Si la condition d'invariance (2.3) est vérifiée, il suffit pour définir une distance géodésique sur $\mathcal{A} = G \times \mathcal{I}$, de définir la famille de normes $(\|\cdot\|_a, a \in \mathcal{A})$ pour tout a de la forme (e, I) où e est l'élément neutre du groupe G .

Démonstration. En exprimant la condition d'invariance (2.3), on peut écrire pour tout $t_0 \in [0; 1]$, tout h dans G et tout chemin (\mathbf{g}, \mathbf{I}) sur $G \times \mathcal{I}$ tel que $(\mathbf{g}, \mathbf{I})(t_0) = (g, I)$:

$$\left\| \frac{d(\mathbf{g}, \mathbf{I})}{dt}(t_0) \right\|_{(g, I)} = \left\| d_{(g, I)} R_h \cdot \frac{d(\mathbf{g}, \mathbf{I})}{dt}(t_0) \right\|_{(g, I).h}$$

On conclut en spécialisant cette relation à $h = g^{-1}$. \square

Remarquons que ces normes dépendent néanmoins de $I \in \mathcal{I}$. Considérons alors un chemin différentiable $\mathbf{a} = (\mathbf{g}, \mathbf{I})$ et plaçons-nous en un temps t_0 tel que $\mathbf{a}(t_0) = (g, I)$. On alors d'après la proposition d'invariance (Prop. 2.10) :

$$\left\| \frac{d\mathbf{a}}{dt}(t_0) \right\|_{g, I} = \left\| d_{\mathbf{a}(t_0)} R_{g^{-1}} \cdot \frac{d\mathbf{a}}{dt}(t_0) \right\|_{e, g, I} \quad (2.5)$$

L'énergie du chemin \mathbf{a} (Déf. 2.7) s'écrit alors :

$$E(\mathbf{a}) = \int_0^1 \left\| d_{\mathbf{a}(t)} R_{\mathbf{g}(t)^{-1}} \cdot \frac{d\mathbf{a}}{dt} \right\|_{e, \mathbf{g}(t), \mathbf{I}(t)}^2 dt$$

Par la suite, on fera l'hypothèse suivante sur les norme $\|\cdot\|_{e, I}$.

Hypothèse 2.12. *On suppose que les normes $\|\cdot\|_{e,I}$ sur $T_{e,I}\mathcal{A} = \mathfrak{G} \times T_I\mathcal{I}$ sont définies de la manière suivante :*

$$\forall \alpha = (v, \delta) \in T_{e,I}\mathcal{A} = \mathfrak{G} \times T_I\mathcal{I}, \|\alpha\|_{e,I}^2 = N_G(v; I)^2 + N_I(\delta; I)^2$$

où $N_G(\cdot; I)$ et $N_I(\cdot; I)$ sont des familles de normes indexées sur \mathcal{I} respectivement sur \mathfrak{G} et $T_I\mathcal{I}$.

On peut alors donner l'expression de l'énergie d'un chemin (\mathbf{g}, \mathbf{I}) sur $G \times \mathcal{I}$ en fonction des normes $N_G(\cdot; I)$ et $N_I(\cdot; I)$ en utilisant les définitions de Déf. 2.5 :

$$\begin{aligned} E(\mathbf{g}, \mathbf{I}) &= \int_0^1 N_G \left(d_{\mathbf{g}(t)} R_{\mathbf{g}(t)}^G \cdot \frac{d\mathbf{g}}{dt}; \mathbf{g}(t). \mathbf{I}(t) \right)^2 dt \\ &+ \int_0^1 N_I \left(d_{\mathbf{I}(t)} R_{\mathbf{g}(t)}^I \cdot \frac{d\mathbf{I}}{dt}; \mathbf{g}(t). \mathbf{I}(t) \right)^2 dt \end{aligned} \quad (2.6)$$

2.2.3 Vitesse et chemin effectif

Définition 2.13. *On définit la vitesse \mathbf{v} d'un chemin de déformation \mathbf{g} sur G comme un chemin sur \mathfrak{G} défini par la relation*

$$\frac{d\mathbf{g}}{dt} = d_e R_{\mathbf{g}(t)}^G \cdot \mathbf{v}(t) \quad (2.7)$$

L'équation (2.7) est appelée équation de transport. La donnée d'un chemin de déformation \mathbf{g} est équivalente à la donnée de sa vitesse \mathbf{v} et de la condition initiale $\mathbf{g}(0) = g$ puisque \mathbf{g} est solution de l'équation différentielle (2.7) sous réserve d'existence et d'unicité de la solution \mathbf{g} à \mathbf{v} donné.

Définition 2.14. *On définit le chemin effectif \mathbf{J} sur \mathcal{I} associé à un chemin $\mathbf{a} = (\mathbf{g}, \mathbf{I})$ sur \mathcal{A} comme le chemin défini par la relation :*

$$\mathbf{J}(t) = \mathbf{g}(t). \mathbf{I}(t) \quad (2.8)$$

Le chemin effectif est donc la résultante de l'action combinée d'un chemin sur le groupe des déformations et d'un chemin sur l'espace des objets.

La donnée du couple de chemins (\mathbf{g}, \mathbf{I}) est donc équivalente à la donnée du couple de chemins (\mathbf{v}, \mathbf{J}) plus la condition initiale $\mathbf{g}(0)$. Mais comme la distance D est invariante par action du groupe G , on ne s'intéressera dans la pratique qu'à des chemins tels que $\mathbf{g}(0) = e$. Il faut aussi noter qu'on a équivalence des chemins (\mathbf{g}, \mathbf{I}) et (\mathbf{v}, \mathbf{J}) sous réserve d'existence et d'unicité de l'équation de transport (2.7).

Proposition 2.15. *L'énergie d'un chemin $\mathbf{a} = (\mathbf{g}, \mathbf{I})$ peut s'exprimer en fonction de sa vitesse \mathbf{v} et de son chemin effectif \mathbf{J} :*

$$E(\mathbf{J}, \mathbf{v}) = \int_0^1 N_G(\mathbf{v}(t); \mathbf{J}(t))^2 dt + \int_0^1 N_I \left(\frac{d\mathbf{J}}{dt} - d_e R_{\mathbf{J}(t)}^I \cdot \mathbf{v}(t); \mathbf{J}(t) \right)^2 dt \quad (2.9)$$

Démonstration. Afin de prouver l'égalité (2.9), on va partir de l'expression (2.6).

Comme on a l'identité $R_g^G \circ R_{g^{-1}}^G = Id$ pour tout $g \in G$, on peut écrire $d_e R_g^G \circ d_g R_{g^{-1}}^G = Id$. On peut réécrire cette identité pour $g = \mathbf{g}(t)$ et l'appliquer à $\frac{d\mathbf{g}}{dt}$, ce qui nous donne :

$$d_e R_{\mathbf{g}(t)}^G \cdot d_{\mathbf{g}(t)} R_{\mathbf{g}(t)}^G \cdot \frac{d\mathbf{g}}{dt} = \frac{d\mathbf{g}}{dt}$$

Ceci nous permet de conclure d'après la définition de la vitesse du chemin \mathbf{g} (cf. Déf. 2.13) que :

$$d_{\mathbf{g}(t)}R_{\mathbf{g}(t)}^G \cdot \frac{d\mathbf{g}}{dt} = \mathbf{v}(t) \quad (2.10)$$

Si on différencie en t l'expression (2.8) du chemin effectif, on obtient :

$$\begin{aligned} \frac{d\mathbf{J}}{dt} &= d_{\mathbf{I}(t)}R_{\mathbf{g}(t)}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} + d_{\mathbf{g}(t)}R_{\mathbf{I}(t)}^{\mathcal{I}} \cdot \frac{d\mathbf{g}}{dt} \\ &= d_{\mathbf{I}(t)}R_{\mathbf{g}(t)}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} + d_{\mathbf{g}(t)}R_{\mathbf{I}(t)}^{\mathcal{I}} \cdot d_e R_{\mathbf{g}(t)}^G \cdot \mathbf{v}(t) \end{aligned}$$

Or on sait que pour tout $g \in G$ et $I \in \mathcal{I}$, on a $R_I^{\mathcal{I}} \circ R_g^G = R_{g.I}^{\mathcal{I}}$. En différenciant cette dernière relation, on obtient $d_g R_I^{\mathcal{I}} \circ d_e R_g^G = d_e R_{g.I}^{\mathcal{I}}$. En spécialisant cette expression à $g = \mathbf{g}(t)$ et $I = \mathbf{I}(t)$, on a :

$$d_{\mathbf{g}(t)}R_{\mathbf{I}(t)}^{\mathcal{I}} \circ d_e R_{\mathbf{g}(t)}^G = d_e R_{\mathbf{J}(t)}^{\mathcal{I}}$$

et ainsi :

$$d_{\mathbf{I}(t)}R_{\mathbf{g}(t)}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} = \frac{d\mathbf{J}}{dt} - d_e R_{\mathbf{J}(t)}^{\mathcal{I}} \cdot \mathbf{v}(t) \quad (2.11)$$

En reportant (2.10) et (2.11) dans (2.6), on obtient bien l'expression de l'énergie souhaitée. \square

2.3 Les métamorphoses

Les métamorphoses [55] sont une manière différente d'envisager le même problème. L'intérêt de cette approche est qu'il donne une vision plus précise de l'action du groupe de déformations G à un niveau infinitésimal. On montrera qu'à quelques détails près, ces deux approches sont tout à fait équivalentes.

2.3.1 Définition et action de groupe

On considère à nouveau un ensemble d'objets \mathcal{I} sur lequel agit un groupe de déformations G . Les métamorphoses sont également définies comme des chemins sur un groupe G et sur un ensemble d'objets \mathcal{I} .

Définition 2.16. *On appelle métamorphose tout chemin (\mathbf{g}, \mathbf{I}) sur un groupe de déformations G et un espace d'objets \mathcal{I} :*

$$t \in [0; 1] \rightarrow \mathbf{a}(t) = (\mathbf{g}(t), \mathbf{I}(t))$$

avec la condition initiale $\mathbf{g}(0) = e$, e étant l'élément neutre du groupe G . On appelle \mathbf{g} la déformation et \mathbf{I} le modèle associés à la métamorphose.

Avec les mêmes notations que précédemment on voit qu'une métamorphose peut aussi être considéré comme un chemin sur \mathcal{A} avec une contrainte sur la déformation initiale.

Remarque 2.17. *Le terme de "modèle" n'est que la traduction de l'anglais "template" (cf. la théorie des "deformable templates" [31]).*

L'action du groupe G sur \mathcal{I} permet alors de définir le chemin effectif \mathbf{J} généré par les deux types d'évolution (déformation et modèle) :

$$\mathbf{J}(t) = \mathbf{g}(t).\mathbf{I}(t) \quad (2.12)$$

On voit qu'on adopte dans le cadre des métamorphoses la même définition du chemin effectif que dans Déf. 2.14. On peut maintenant se placer à un instant donné t du chemin (\mathbf{g}, \mathbf{I}) et étudier ce qu'il se passe à un niveau infinitésimal.

2.3.2 Actions infinitésimales

Vitesse

Si on suppose que l'application R_g^G est C^1 (cf. Déf. 2.5), on peut définir sa différentielle en l'élément neutre e de G :

$$\begin{aligned} d_e R_g^G : \mathfrak{G} &\rightarrow \mathfrak{G} \\ v &\rightarrow v(g) \end{aligned}$$

L'application $g \rightarrow v(g)$ est donc un champ de vecteurs continu sur G . Elle correspond à l'action infinitésimale d'un élément v de \mathfrak{G} sur une déformation g .

On est alors à même de donner une définition de la vitesse d'un chemin de déformation identique à Déf. 2.13.

Définition 2.18. *La vitesse \mathbf{v} d'un chemin \mathbf{g} sur G est définie par la relation suivante :*

$$\frac{d\mathbf{g}}{dt} = d_e R_{\mathbf{g}(t)}^G \cdot \mathbf{v}(t) = \mathbf{v}(t)(\mathbf{g}(t)) \quad (2.13)$$

Métamorphose infinitésimale

Si on suppose de plus que $R_I^{\mathcal{I}}$ (cf. Déf. 2.5) est C^1 , alors sa différentielle en l'élément neutre e de G est une application de \mathfrak{G} dans l'espace $T_I \mathcal{I}$ tangent à \mathcal{I} en I :

$$\begin{aligned} d_e R_I^{\mathcal{I}} : \mathfrak{G} &\rightarrow T_I \mathcal{I} \\ v &\rightarrow v(I) \end{aligned}$$

L'application $I \rightarrow v(I)$ est donc un champ de vecteurs continu sur \mathcal{I} . Elle correspond à l'action infinitésimale d'un élément v de \mathfrak{G} sur un objet I .

Exemple 2.19. *Si on considère à nouveau le cas de l'exemple 2.3, l'algèbre de Lie associée au groupe G des rotations de \mathbb{R}^d est l'algèbre des endomorphismes antisymétriques de \mathbb{R}^d . L'action infinitésimale du groupe G sur un N -uplet I peut être vue comme l'action d'un endomorphisme antisymétrique sur chacun des points du N -uplet I .*

Définition 2.20. *On appelle métamorphose infinitésimale en $J \in \mathcal{I}$ tout élément $\eta = (v, \delta)$ de $\mathfrak{G} \times T_J \mathcal{I}$, qui génère une déformation infinitésimale $v(J) + \delta$.*

On peut faire le lien entre les métamorphoses et les métamorphoses infinitésimales en différenciant la relation (2.12). On obtient en effet :

$$\begin{aligned} \frac{d\mathbf{J}}{dt} &= d_{\mathbf{I}(t)} R_{\mathbf{g}(t)^{-1}}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} + d_{\mathbf{g}(t)} R_{\mathbf{I}(t)}^{\mathcal{I}} \cdot \frac{d\mathbf{g}}{dt} \\ &= d_{\mathbf{I}(t)} R_{\mathbf{g}(t)^{-1}}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} + d_e R_{\mathbf{g}(t), \mathbf{I}(t)}^{\mathcal{I}} \cdot d_{\mathbf{g}(t)} R_{\mathbf{g}(t)^{-1}}^G \cdot \frac{d\mathbf{g}}{dt} \\ &= d_{\mathbf{I}(t)} R_{\mathbf{g}(t)^{-1}}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} + \mathbf{v}(t)(\mathbf{J}(t)) \end{aligned} \quad (2.14)$$

La métamorphose infinitésimale en $\mathbf{J}(t)$ correspondante est alors :

$$\eta(t) = \left(\mathbf{v}(t), d_{\mathbf{I}(t)} R_{\mathbf{g}(t)^{-1}}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} \right)$$

puisque l'on peut vérifier que $\mathbf{v}(t) \in \mathfrak{G}$ et que $d_{\mathbf{I}(t)} R_{\mathbf{g}(t)^{-1}}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} \in T_{\mathbf{J}(t)}\mathcal{I}$.

On peut alors définir des normes sur les métamorphoses infinitésimales à partir d'une norme $|\cdot|_{\mathfrak{G}}$ sur \mathfrak{G} et de normes $|\cdot|_J$ sur les espaces tangents $T_J\mathcal{I}$.

Proposition 2.21. *L'ensemble $\mathfrak{G} \times T_J\mathcal{I}$ des métamorphoses infinitésimales en J est muni d'une structure hilbertienne en posant pour tout $\eta = (v, \delta) \in \mathfrak{G} \times T_J\mathcal{I}$:*

$$\|\eta\|_{e,J}^2 = |v|_{\mathfrak{G}}^2 + |\delta|_J^2$$

Remarque 2.22. *Les normes $|\cdot|_{\mathfrak{G}}$ et $|\cdot|_J$ correspondent aux normes $N_G(\cdot, J)$ et $N_{\mathcal{I}}(\cdot, J)$ de l'hypothèse 6.1 à la différence près que les normes sur \mathfrak{G} ne sont plus indexées sur \mathcal{I} .*

2.3.3 Construction d'une métrique riemannienne

A partir de la norme sur l'espace des métamorphoses infinitésimales en J (Prop. 2.21), on peut déduire une nouvelle norme sur $T_J\mathcal{I}$.

Proposition 2.23. *La relation suivante pour $j \in T_J\mathcal{I}$:*

$$\begin{aligned} \|j\|_J^2 &= \inf_{v \in \mathfrak{G}} \{ |v|_{\mathfrak{G}}^2 + |\delta|_J^2 : v(J) + \delta = j \} \\ &= \inf_{v \in \mathfrak{G}} \{ |v|_{\mathfrak{G}}^2 + |j - v(J)|_J^2 \} \end{aligned} \quad (2.15)$$

définit une norme sur $T_J\mathcal{I}$.

On se reportera à [55] pour la démonstration de cette proposition.

On peut alors définir une métrique sur \mathcal{I} à partir des métriques sur ses espaces tangents.

Définition 2.24. *L'énergie d'un chemin \mathbf{J} sur \mathcal{I} est donnée par :*

$$E(\mathbf{J}) = \int_0^1 \left\| \frac{d\mathbf{J}}{dt} \right\|_{\mathbf{J}(t)}^2 dt = \inf_{\mathbf{v}} \left\{ \int_0^1 |\mathbf{v}(t)|_{\mathfrak{G}}^2 dt + \int_0^1 \left| \frac{d\mathbf{J}}{dt} - \mathbf{v}(t)(\mathbf{J}(t)) \right|_{\mathbf{J}(t)}^2 dt \right\} \quad (2.16)$$

où \mathbf{v} appartient à l'ensemble des chemins sur \mathfrak{G} .

Donc en posant pour (\mathbf{J}, \mathbf{v}) chemin sur $\mathcal{I} \times \mathfrak{G}$:

$$U(\mathbf{J}, \mathbf{v}) = \int_0^1 |\mathbf{v}(t)|_{\mathfrak{G}}^2 dt + \int_0^1 \left| \frac{d\mathbf{J}}{dt} - \mathbf{v}(t)(\mathbf{J}(t)) \right|_{\mathbf{J}(t)}^2 dt \quad (2.17)$$

on a :

$$E(\mathbf{J}) = \inf_{\mathbf{v}} \{U(\mathbf{J}, \mathbf{v})\}$$

et la distance géodésique entre deux éléments I et I' de \mathcal{I} est donnée par :

$$\begin{aligned} d(I, I') &= \inf \{E(\mathbf{J}) : \mathbf{J}(0) = I, \mathbf{J}(1) = I'\} \\ &= \inf \{U(\mathbf{J}, \mathbf{v}) : \mathbf{J}(0) = I, \mathbf{J}(1) = I'\} \end{aligned}$$

Si on revient maintenant à la définition des métamorphoses (2.12) et que l'on considère le flot de déformations \mathbf{g} généré par le chemin \mathbf{v} par le biais de (2.13), on retrouve la partie déformation de la métamorphose. Et la partie modèle est donnée par $\mathbf{I}(t) = \mathbf{g}(t)^{-1} \cdot \mathbf{J}(t)$.

De plus, les relations (2.13) et (2.14) nous donnent une expression de l'énergie en fonction de \mathbf{I} et \mathbf{g} :

$$U(\mathbf{g}, \mathbf{I}) = \int_0^1 \left| d_{\mathbf{g}(t)} R_{\mathbf{g}(t)^{-1}}^G \cdot \frac{d\mathbf{g}}{dt} \right|_{\mathfrak{G}}^2 dt + \int_0^1 \left| d_{\mathbf{I}(t)} R_{\mathbf{I}(t)^{-1}}^I \cdot \frac{d\mathbf{I}}{dt} \right|_{\mathbf{J}(t)}^2 dt \quad (2.18)$$

2.3.4 Comparaison avec l'approche précédente

On a donc vu dans cette section sur les métamorphoses que les chemins \mathbf{g} , \mathbf{I} , \mathbf{v} , \mathbf{J} avaient exactement la même signification que dans l'approche initiale. De plus, on constate que les énergies des chemins (\mathbf{g}, \mathbf{I}) (2.18) et (\mathbf{v}, \mathbf{J}) (2.17) ont exactement la même expression que dans (2.6) et (2.9). La seule différence entre les deux approches est signalée dans Rem. 2.22.

2.4 Exemples

On proposera deux applications de la théorie introduite plus haut et qu'on utilisera à nouveau par la suite : la mise en correspondance d'images et la mise en correspondance de points.

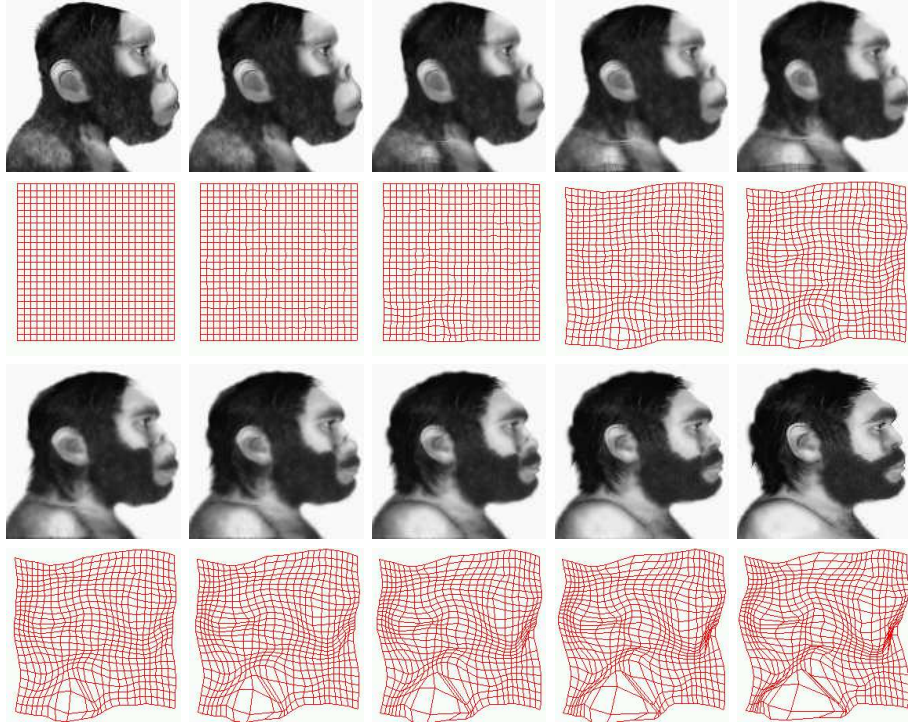
2.4.1 Mise en correspondance d'images

Définitions

Soit Ω un ouvert borné de \mathbb{R}^2 ¹. Une image I est modélisée comme une application de Ω dans \mathcal{R} . On appelle \mathcal{I} l'espace fonctionnel contenant les images qui nous intéressent. Soit G le groupe des difféomorphismes de Ω . G définit une action de groupe à droite sur \mathcal{I} de la manière suivante :

$$\begin{aligned} G \times \mathcal{I} &\rightarrow \mathcal{I} \\ (g, I) &\rightarrow I.g = I \circ g^{-1} \end{aligned}$$

¹Tout ce qui suit reste cependant valable pour des dimensions supérieures à 2 (mise en correspondance de volumes par exemple).

FIG. 2.2 – Chemin \mathbf{J} d'images et chemin \mathbf{g} des difféomorphismes associés

Autrement dit, le groupe G agit sur \mathcal{I} par l'intermédiaire d'un changement de paramétrage. De plus, G agit naturellement sur lui-même :

$$\begin{aligned} G \times G &\rightarrow G \\ (g, h) &\rightarrow g.h = g \circ h \end{aligned}$$

L'algèbre de Lie \mathfrak{G} associée à G peut être identifiée à un sous-espace des champs de vecteurs continus sur Ω .

La figure 2.2 donne un exemple de chemin \mathbf{I} d'une image vers une autre ainsi que le chemin \mathbf{g} de difféomorphismes associés (ici représentés par des grilles).

Détermination de la distance géodésique

On pose pour un chemin $\mathbf{a} = (\mathbf{g}, \mathbf{I})$ sur \mathcal{A} :

$$\frac{d\mathbf{a}}{dt} = \left(\frac{\partial \mathbf{g}}{\partial t}, \frac{\partial \mathbf{I}}{\partial t} \right)$$

Et un calcul immédiat nous donne :

$$d_{\mathbf{a}(t_0)} R_{g^{-1}} \cdot \frac{d\mathbf{a}}{dt}(t_0) = \left(\frac{\partial \mathbf{g}}{\partial t}(t_0, \cdot) \circ g^{-1}, \frac{\partial \mathbf{I}}{\partial t}(t_0, \cdot) \circ g^{-1} \right) \quad (2.19)$$

On définit la famille de normes $\{\|\cdot\|_a, a \in \mathcal{A}\}$ à partir de Prop. 2.11 et de l'hypothèse 2.12. Dans le cas général où \mathbf{a} est un chemin tel que $\mathbf{a}(t_0) = (g, I)$,

on peut donc écrire d'après (2.19) :

$$\begin{aligned} \left\| \frac{d\mathbf{a}}{dt}(t_0) \right\|_{g,I}^2 &= N_G \left(\frac{\partial \mathbf{g}}{\partial t}(t_0, \cdot) \circ g^{-1}; I \circ g^{-1} \right)^2 \\ &\quad + N_{\mathcal{I}} \left(\frac{\partial \mathbf{I}}{\partial t}(t_0, \cdot) \circ g^{-1}; I \circ g^{-1} \right)^2 \end{aligned} \quad (2.20)$$

Et l'énergie associée au chemin \mathbf{a} (2.6) s'écrit :

$$\begin{aligned} E(\mathbf{a}) &= \int_0^1 N_G \left(\frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{g}^{-1}(t, \cdot)); \mathbf{I}(t, \mathbf{g}^{-1}(t, \cdot)) \right)^2 dt \\ &\quad + \int_0^1 N_{\mathcal{I}} \left(\frac{\partial \mathbf{I}}{\partial t}(t, \mathbf{g}^{-1}(t, \cdot)); \mathbf{I}(t, \mathbf{g}^{-1}(t, \cdot)) \right)^2 dt \end{aligned} \quad (2.21)$$

où l'on note $\mathbf{g}^{-1}(t) = \mathbf{g}(t)^{-1}$.

On peut réexprimer l'énergie cette énergie de manière différente en utilisant la vitesse \mathbf{v} et le chemin effectif \mathbf{J} comme on l'a fait dans 2.2.3. Dans le cas présent, la vitesse et le chemin effectif s'expriment comme suit.

$$\mathbf{v}(t, x) = \frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{g}^{-1}(t, x)) \quad (2.22)$$

$$\mathbf{J}(t, x) = \mathbf{I}(t, \mathbf{g}^{-1}(t, x)) \quad (2.23)$$

Autrement dit, on adopte passe des coordonnées lagrangiennes aux coordonnées eulériennes.

Si \mathbf{I} et par conséquent \mathbf{J} sont différentiables en la variable spatiale x , on peut dériver l'identité $\mathbf{I}(t, y) = \mathbf{J}(t, \mathbf{g}(t, y))$ par rapport au temps, ce qui nous donne :

$$\frac{\partial \mathbf{I}}{\partial t}(t, y) = \frac{\partial \mathbf{J}}{\partial t}(t, \mathbf{g}(t, y)) + \frac{\partial \mathbf{J}}{\partial x}(t, \mathbf{g}(t, y)) \cdot \frac{\partial \mathbf{g}}{\partial t}(t, y)$$

En prenant $x = \mathbf{g}(t, y)$, on arrive finalement à l'expression :

$$\frac{\partial \mathbf{I}}{\partial t}(t, \mathbf{g}^{-1}(t, x)) = \frac{\partial \mathbf{J}}{\partial t}(t, x) + \frac{\partial \mathbf{J}}{\partial x}(t, x) \cdot \mathbf{v}(t, x)$$

Ceci n'est autre que la dérivée de Lie (ou dérivée particulière) de \mathbf{J} suivant le champ \mathbf{v} .

L'énergie du chemin $\mathbf{a} = (\mathbf{g}, \mathbf{I})$ s'exprime alors de la manière suivante en fonction de \mathbf{J} et \mathbf{v} :

$$E(\mathbf{J}, \mathbf{v}) = \int_0^1 N_G(\mathbf{v}(t, \cdot); \mathbf{J}(t, \cdot))^2 dt + \int_0^1 N_{\mathcal{I}} \left(\frac{\partial \mathbf{J}}{\partial t}(t, \cdot) + \frac{\partial \mathbf{J}}{\partial x} \cdot \mathbf{v}(t, \cdot); \mathbf{J}(t, \cdot) \right)^2 dt \quad (2.24)$$

On aurait pu obtenir directement cette expression à partir de la formule (2.9) mais il aurait fallu donner un sens à la différentielle de l'application $R_{\mathcal{I}}^{\mathcal{I}} : g \rightarrow I \circ g^{-1}$. Une différentiation formelle de cette application en l'élément neutre e donne $d_e R_{\mathcal{I}}^{\mathcal{I}}(v) = -\langle \nabla I, v \rangle$ mais il faut être plus soigneux si l'on veut une démonstration exacte. On peut trouver dans [62] des conditions générales sur l'ensemble des images \mathcal{I} pour que la différentiation de $g \rightarrow g.I = I \circ g^{-1}$ soit licite.

Le premier terme de l'énergie (2.24) sur le chemin de champs de vecteurs \mathbf{v} permet d'imposer des contraintes de régularité sur \mathbf{v} et, par conséquent sur le difféomorphisme g généré par ce champ de déformations par le biais de l'équation de transport (2.7). Le second terme impose la conservation du flot optique généré par le champ \mathbf{v} . Finalement, il s'agit de calculer le flot optique avec régularisation [33] mais dans un cadre de *grandes* déformations.

Nous verrons dans le chapitre 3 comment minimiser cette énergie.

Lien avec les métamorphoses

On peut faire le lien avec les métamorphoses. Ici \mathbf{I} serait la partie modèle de la métamorphose et \mathbf{g} la partie déformation.

Comme on l'a dit précédemment, en différenciant formellement $g \rightarrow g.I = I \circ g^{-1}$ en l'identité, on obtient l'expression de l'action infinitésimale d'un élément v de \mathfrak{G} sur une image J :

$$v(J) = -\langle \nabla J, v \rangle$$

si bien qu'en remplaçant dans (2.17), on retrouve la même expression de l'énergie que dans (2.24).

2.4.2 Mise en correspondance de points

Cette application a déjà été traitée dans [61] sous l'appellation de splines géodésiques ou auparavant dans un cadre théorique légèrement différent [43, 16, 35, 36] (on ne s'y intéresse pas à l'action de groupe de déformations). On en donnera une extension dans le chapitre 6.

Définitions

Soit G un groupe de difféomorphismes de Ω ouvert de \mathcal{R}^d et $\mathcal{I} = \Omega^n$ i.e. les n -uplets de points. On définit l'action de G sur \mathcal{I} de la manière suivante. Un élément g de G agit à droite sur un élément $I = (p_1, \dots, p_n)$ de \mathcal{I} de la manière suivante :

$$I.g = g^{-1}(I) = (g^{-1}(p_1), \dots, g^{-1}(p_n))$$

L'action naturelle à droite de G sur lui-même est encore cette fois-ci :

$$\forall g, h \in G, g.h = g \circ h$$

Détermination de la distance géodésique

On se donne maintenant un chemin sur \mathcal{A} .

$$\mathbf{a}(t) = (\mathbf{g}(t, \cdot), \mathbf{I}(t)) = (\mathbf{g}(t, \cdot), \mathbf{p}_1(t), \dots, \mathbf{p}_n(t)) \quad (2.25)$$

On reprend alors le même raisonnement que dans 2.4.1. En gardant les mêmes notations que dans l'exemple des images, on a donc encore la relation (2.5) et on peut écrire une expression de la différentielle de l'application de translation similaire à (2.19) :

$$d_{\mathbf{a}(t_0)} R_{g^{-1}} \cdot \frac{d\mathbf{a}}{dt}(t_0) = \left(\frac{\partial \mathbf{g}}{\partial t}(t_0, \mathbf{g}^{-1}(t_0, \cdot)), d_{\mathbf{p}_1(t_0)} g \cdot \frac{d\mathbf{p}_1}{dt}, \dots, d_{\mathbf{p}_n(t_0)} g \cdot \frac{d\mathbf{p}_n}{dt} \right) \quad (2.26)$$

avec $\mathbf{a}(t_0) = (g, I)$.

On fait à nouveau le même type de changement de variable que dans (2.22) et (2.23) :

$$\mathbf{v}(t, x) = \frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{g}^{-1}(t, x)) \quad (2.27)$$

$$\mathbf{q}_i(t) = \mathbf{g}(t, \mathbf{p}_i(t)) \quad (2.28)$$

Pour garder la cohérence avec les notations de l'exemple précédent, on pourra noter $\mathbf{J}(t) = (\mathbf{q}_1(t), \dots, \mathbf{q}_n(t))$. En dérivant l'identité (2.28) par rapport au temps, on obtient :

$$\begin{aligned} \frac{d\mathbf{q}_i}{dt} &= \frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{p}_i(t)) + d_{\mathbf{p}_i(t)}\mathbf{g}(t) \cdot \frac{d\mathbf{p}_i}{dt} \\ &= \mathbf{v}(t, \mathbf{q}_i(t)) + d_{\mathbf{p}_i(t)}\mathbf{g}(t) \cdot \frac{d\mathbf{p}_i}{dt} \end{aligned}$$

De la même manière qu'on a, pour un difféomorphisme g de G et un n -uplet $I = (p_1, \dots, p_n)$ de \mathcal{I} , $g(I) = (g(p_1), \dots, g(p_n))$, on notera aussi, pour un champ de vecteurs v sur Ω , $v(I) = (v(p_1), \dots, v(p_n))$.

On définit à nouveau des familles de normes N_G et $N_{\mathcal{I}}$ comme dans l'hypothèse 2.12, ce qui nous donne :

$$\begin{aligned} \left\| \frac{d\mathbf{a}}{dt}(t_0) \right\|_{g, I}^2 &= \left\| d_{\mathbf{a}(t_0)}R_{g^{-1}} \cdot \frac{d\mathbf{a}}{dt}(t_0) \right\|_{Id, g^{-1}.I}^2 \\ &= N_G(\mathbf{v}(t_0, \cdot); \mathbf{J}(t_0))^2 + N_{\mathcal{I}} \left(\frac{d\mathbf{J}}{dt}(t_0) - \mathbf{v}(t_0, \mathbf{J}(t_0)); \mathbf{J}(t_0) \right)^2 \end{aligned}$$

On en tire donc une expression de l'énergie :

$$E(\mathbf{J}, \mathbf{v}) = \int_0^1 N_G(\mathbf{v}(t, \cdot); \mathbf{J}(t))^2 dt + \int_0^1 N_{\mathcal{I}} \left(\frac{d\mathbf{J}}{dt} - \mathbf{v}(t, \mathbf{J}(t)); \mathbf{J}(t) \right)^2 dt \quad (2.29)$$

Encore une fois, le premier terme de cette énergie sert à imposer une certaine régularité au champ \mathbf{v} tandis que le second terme mesure l'adéquation entre les trajectoire des landmarks et le difféomorphisme en comparant la vitesse de chaque trajectoire avec le champ de vitesse \mathbf{v} généré par le difféomorphisme en un même point comme on peut le voir dans la figure 2.3.

Nous verrons comment minimiser cette énergie et ainsi trouver les géodésiques entre deux ensembles de landmarks dans l'extension présentée au chapitre 6.

Lien avec les métamorphoses

On peut encore une fois faire le lien avec les métamorphoses. Dans le contexte de la mise en correspondance de points, les \mathbf{p}_i représenteraient la partie modèle de la métamorphose et \mathbf{g} la partie déformation. Si on pose $J = (q_1, \dots, q_n)$, en différenciant $g \rightarrow g.J = g(J)$ en l'identité, on obtient :

$$v(J) = (v(q_1), \dots, v(q_n))$$

ce qui donne la même expression que l'énergie (2.29) en remplaçant à nouveau dans (2.17).

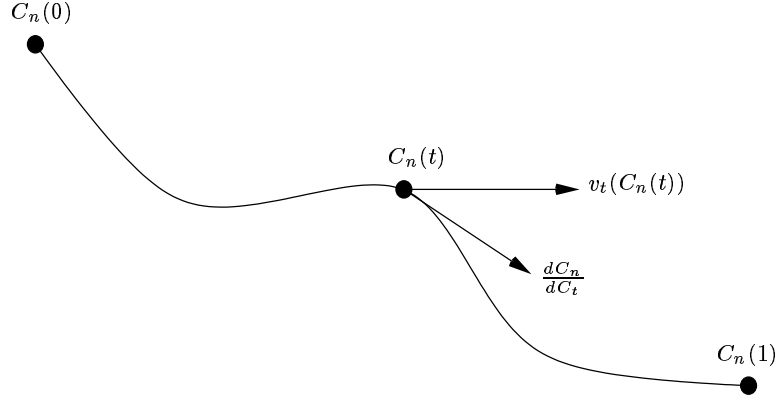


FIG. 2.3 – Trajectoire d'un point

2.5 Existence d'un minimum de l'énergie

On formule tout d'abord l'hypothèse suivante qui sera respectée dans toutes les applications qui suivront.

Hypothèse 2.25. *On suppose que les normes $N_{\mathfrak{G}}(\cdot; I)$ ne sont pas indexées sur \mathcal{I} . On suppose de plus que chaque espace tangent $T_I\mathcal{I}$ peut être plongé dans un même espace de Banach \mathcal{Z} de sorte que les normes $N_{\mathcal{I}}(\cdot, I)$ sont prises comme la norme sur \mathcal{Z} et ne sont donc plus indexées sur \mathcal{I} .*

On va associer à tout chemin (\mathbf{g}, \mathbf{I}) le chemin (\mathbf{v}, \mathbf{Z}) tel que :

$$\begin{cases} \mathbf{v}(t) = d_{\mathbf{g}(t)}R_{\mathbf{g}(t)}^G \cdot \frac{d\mathbf{g}}{dt} \\ \mathbf{Z}(t) = d_{\mathbf{I}(t)}R_{\mathbf{g}(t)}^{\mathcal{I}} \cdot \frac{d\mathbf{I}}{dt} \end{cases}$$

La donnée du chemin (\mathbf{g}, \mathbf{I}) est donc équivalente à la donnée de (\mathbf{v}, \mathbf{Z}) sous réserve que le système d'équations différentielles suivant admette une unique solution (\mathbf{g}, \mathbf{J}) à (\mathbf{v}, \mathbf{Z}) fixé :

$$\begin{cases} \frac{d\mathbf{g}}{dt} = d_e R_{\mathbf{g}(t)}^G \cdot \mathbf{v}(t) \\ \frac{d\mathbf{I}}{dt} = d_{\mathbf{g}(t), \mathbf{I}(t)} R_{\mathbf{g}(t)}^{\mathcal{I}} \cdot \mathbf{Z}(t) \end{cases} \quad (2.30)$$

avec les conditions initiales $\mathbf{g}(0) = e$ et $\mathbf{I}(0) = I_0$.

Définition 2.26. *On note \mathcal{K} l'espace de Hilbert des chemins (\mathbf{v}, \mathbf{Z}) sur $\mathfrak{G} \times \mathcal{Z}$ qui vérifient :*

$$\|(\mathbf{v}, \mathbf{Z})\|_{\mathcal{K}}^2 = \int_0^1 N_G(\mathbf{v}(t))^2 dt + \int_0^1 N_{\mathcal{I}}(\mathbf{Z}(t))^2 dt < \infty$$

On note $(\mathbf{g}_{\mathbf{v}}, \mathbf{I}_{\mathbf{v}, \mathbf{Z}})$ la solution de (2.30) si elle existe.

Plutôt que de prouver l'existence d'un chemin (\mathbf{g}, \mathbf{I}) minimisant l'énergie (2.6) sous les conditions $\mathbf{g}(0) = e$, $\mathbf{I}(0) = I_0$ et $\mathbf{g}(1) \cdot \mathbf{I}(1) = I_1$, on va prouver

l'existence d'un chemin (\mathbf{v}, \mathbf{Z}) minimisant la norme $\|\cdot\|_{\mathcal{K}}$ sous les conditions $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(0) = I_0$ et $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(1) = I_1$.

La fonctionnelle $\|\cdot\|_{\mathcal{K}}$ est convexe et continue pour la topologie forte donc elle est aussi semi-continue inférieurement pour la topologie faible.

Notons $\mathcal{V}(I_0, I_1)$, pour $I_0, I_1 \in \mathcal{I}$ le sous-ensemble de \mathcal{K} des couples (\mathbf{v}, \mathbf{Z}) tels que $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(0) = I_0$ et $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(1) = I_1$. On note aussi $B_{\mathcal{K}}(R)$ la boule centrée fermée de \mathcal{K} de rayon R .

Lemme 2.27. *Pour tout $R < \infty$ et tout $I_0, I_1 \in \mathcal{I}$, $\mathcal{V}(I_0, I_1) \cap B_{\mathcal{K}}(R)$ est faiblement compact.*

Démonstration. On sait que les boules fermées de rayon fini sont compactes pour la topologie faible. Reste à prouver que si une suite $(\mathbf{v}_n, \mathbf{Z}_n)$ de $\mathcal{V}(I_0, I_1) \cap B_{\mathcal{K}}(R)$ tend faiblement vers (\mathbf{v}, \mathbf{Z}) alors on a bien $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(0) = I_0$ et $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(1) = I_1$. Soit donc une telle suite $(\mathbf{v}_n, \mathbf{Z}_n)$. On peut alors prouver sous certaines conditions [42] que $\mathbf{g}_{\mathbf{v}_n}$ et $\mathbf{I}_{\mathbf{v}_n, \mathbf{Z}_n}$ convergent uniformément sur $[0; 1]$ vers $\mathbf{g}_{\mathbf{v}}$ et $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}$. En particulier, on a bien $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(0) = I_0$ et $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(1) = I_1$. \square

Grâce à ce lemme et à la semi-continuité faible de la norme $\|\cdot\|_{\mathcal{K}}$ on peut donc conclure qu'il existe un couple (\mathbf{v}, \mathbf{Z}) minimisant la norme $\|\cdot\|_{\mathcal{K}}$ sous les conditions $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(0) = I_0$ et $\mathbf{I}_{\mathbf{v}, \mathbf{Z}}(1) = I_1$.

Chapitre 3

Mise en correspondance d'images et de modèles numériques d'élévation

3.1 Déformations dans le cadre de la détection de changement

On reprend dans ce chapitre la mise en correspondance d'images introduite dans 2.4.1. On considérera aussi les MNE comme des images, la seule différence venant du fait que la valeur de chaque pixel correspond à une altitude et non plus à un niveau de gris. Pour plus de précisions sur la façon dont les MNE sont construits, on pourra se reporter à l'annexe A. On s'attachera dans ce chapitre à définir des normes de régularisation des déformations adéquates ainsi que des stratégies permettant d'atteindre le minimum global de l'énergie (2.24).

3.1.1 Choix de l'énergie et des inconnues

On utilise ce modèle de déformation pour pouvoir détecter des changements dans les trois dimensions. En effet, il est important de pouvoir détecter les changements à la fois en planimétrie et en altimétrie ou en niveaux de gris (cf. fig. 3.1).

Nous choisirons les normes fonctionnelles N_G et N_I de l'équation (2.24) de manière à ce qu'elles ne dépendent pas de la variable \mathbf{J} . Plus précisément, nous prendrons $N_G(\cdot) = \|\cdot\|_{\mathcal{V}}$ où \mathcal{V} est un Hilbert inclus dans $L^2(\Omega, \mathbb{R}^2)$ tel que $\|\cdot\|_{\mathcal{V}}$ soit une norme régularisante (e.g. \mathcal{V} est un espace de Sobolev) et $N_I(\cdot) = \|\cdot\|_{L^2}$.

$$E(\mathbf{J}, \mathbf{v}) = \lambda \int_0^1 \|\mathbf{v}(t, \cdot)\|_{\mathcal{V}}^2 dt + \int_0^1 \int_{\Omega} \left| \langle \nabla \mathbf{J}(t, x), \mathbf{v}(t, x) \rangle + \frac{\partial \mathbf{J}}{\partial t}(t, x) \right|^2 dx dt \quad (3.1)$$

Le facteur λ permet de régler l'équilibre entre la régularité de la déformation et la conservation du flot optique. Suivant la valeur de λ , l'algorithme se comporte de manière différente comme l'illustre la figure 3.2 :

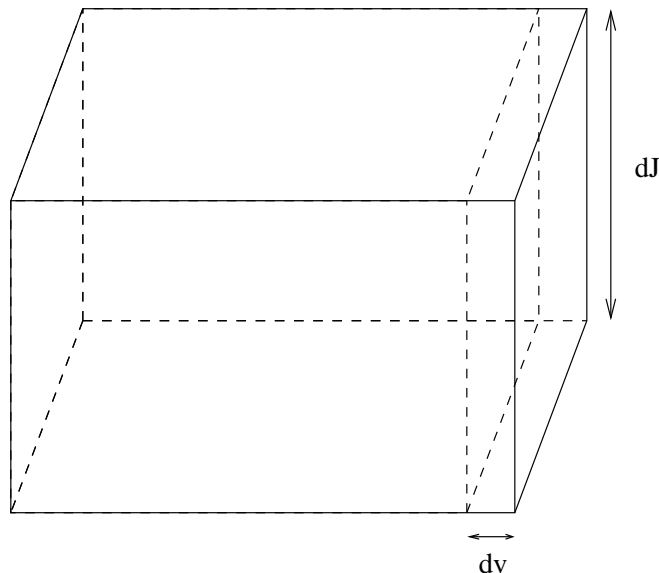


FIG. 3.1 – Deux possibilités : modifier l’altimétrie ou la planimétrie

- $\lambda \gg 1$: l’algorithme minimise les déformations en planimétrie (cf. fig 3.2(a))
- en déformation pure avec la terminologie des métamorphoses (cf. 2.3) ,
- $\lambda \ll 1$: l’algorithme minimise les changements en altimétrie ou en niveaux de gris (cf. fig. 3.2(b)) – en template avec la terminologie des métamorphoses.

Dans le cas de la détection de changements, on a affaire à deux types de déformations :

- les déformations dues à l’apparition de changements (e.g. apparition d’un nouveau bâtiment) qui correspondront plutôt à des déformations altimétriques,
- et les déformations dues à l’imprécision des deux MNE comparés (imprécisions des contours de bâtiments notamment) qui correspondront plutôt à des déformations planimétriques.

3.1.2 Régularisation par un noyau gaussien

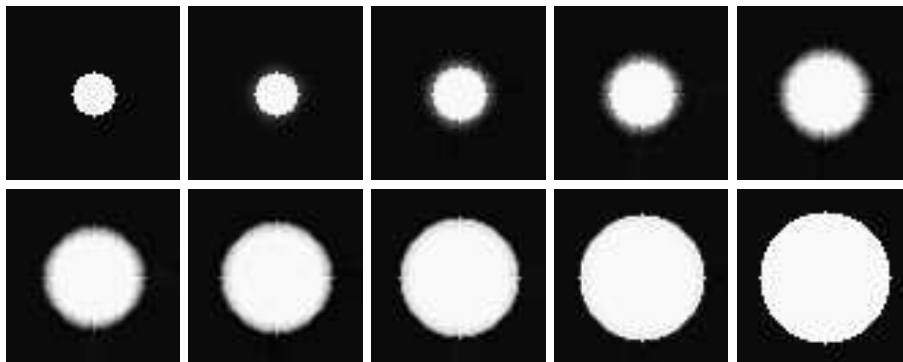
La première idée venant à l’esprit pour régulariser le champ de déformation \mathbf{v} est d’utiliser une régularisation gaussienne. On considère donc que \mathcal{V} est un espace de Hilbert à noyau reproduisant N^2 (cf. appendice B) où N (et par conséquent N^2) est un noyau gaussien.

Si on pose $\mathbf{v} = N\mathbf{w}$, on a alors :

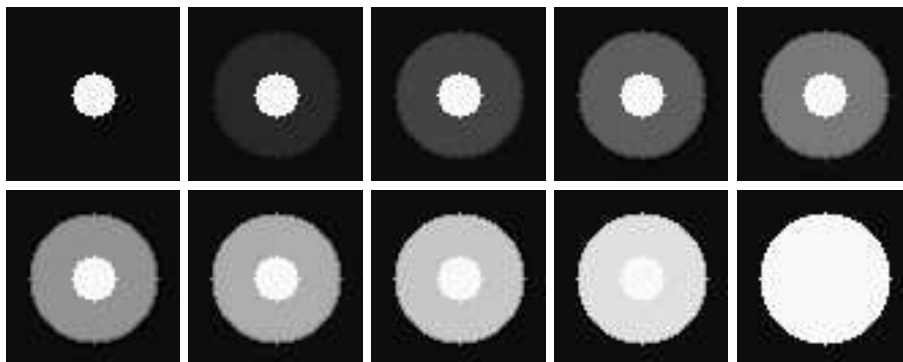
$$E(\mathbf{J}, \mathbf{w}) = \lambda \int_0^1 \int_{\Omega} |\mathbf{w}(t, x)|^2 dx dt + \int_0^1 \int_{\Omega} \left| \langle \nabla \mathbf{J}(t, x), N\mathbf{w}(t, x) \rangle + \frac{\partial \mathbf{J}}{\partial t}(t, x) \right|^2 dx dt \quad (3.2)$$

On peut voir le résultat de la minimisation d’une énergie de ce type dans le cadre de la mise en correspondance d’images (cf. fig. 3.3).

La taille du noyau gaussien influe sur le résultat final. En effet, plus le noyau est grand, plus il est rigide : la déformation est plus régulière. Néanmoins, un



(a) Déformations en planimétrie : $\lambda \gg 1$



(b) Déformations en niveaux de gris : $\lambda \ll 1$

FIG. 3.2 – Influence du poids de régularisation λ sur la mise en correspondance du petit disque et du grand disque

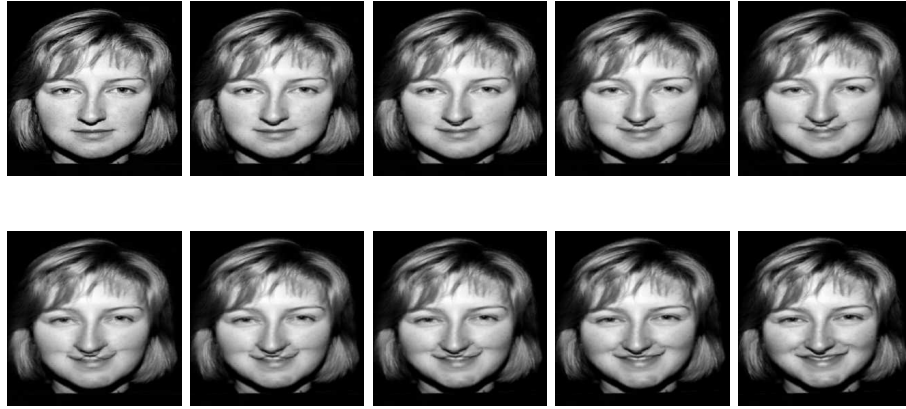


FIG. 3.3 – Morphing avec un opérateur de régularisation gaussien. Les images entre la première et la dernière images sont calculées par l’algorithme.

noyau plus grand permet de retrouver de plus grandes déformations comme on peut le constater en comparant les résultats de la figure 3.4. On constate bien que la déformation est plus régulière avec le noyau le plus grand ($\sigma = 10$ pixels) qu’avec les deux autres. On s’aperçoit aussi qu’avec le plus petit noyau ($\sigma = 3$ pixels), on n’arrive pas à retrouver une déformation “raisonnable”.

3.1.3 Décomposition de la déformation dans une base appropriée

Dans certains cas, la régularisation gaussienne ne permet pas de retrouver de grandes déformations quelque soit la taille du noyau employé. La figure 3.5(a) montre le meilleur résultat que l’on ait pu obtenir avec la régularisation gaussienne.

Dans ce cas, on définit l’espace de Hilbert \mathcal{V} à partir de sa norme :

$$\|\cdot\|_{\mathcal{V}} = \|R\cdot\|_{L^2}$$

où R est un opérateur tel que la norme $\|\cdot\|_{\mathcal{V}}$ soit régularisante. Par exemple, on peut envisager de prendre pour R un opérateur différentiel.

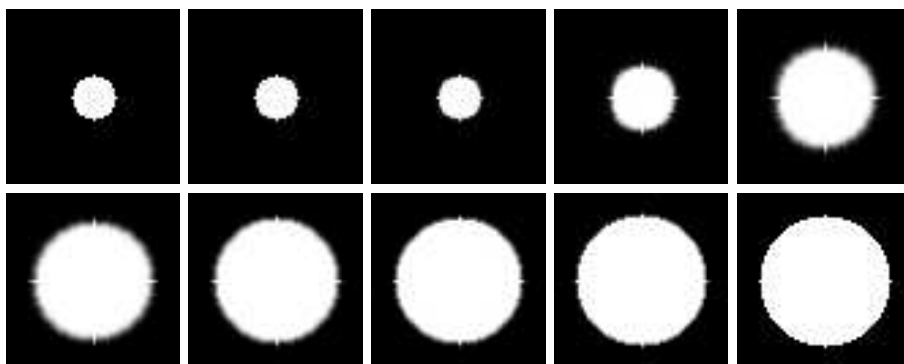
En fait, on va tenter de décomposer la déformation dans une base de fonctions de \mathcal{V} appropriée et définir l’opérateur R à partir de cette décomposition. Pour être plus précis, si on adopte une base orthonormale de fonctions $\{e_i\}$ dans laquelle on décompose le champ de déformation v i.e.

$$v = \sum_i v_i e_i$$

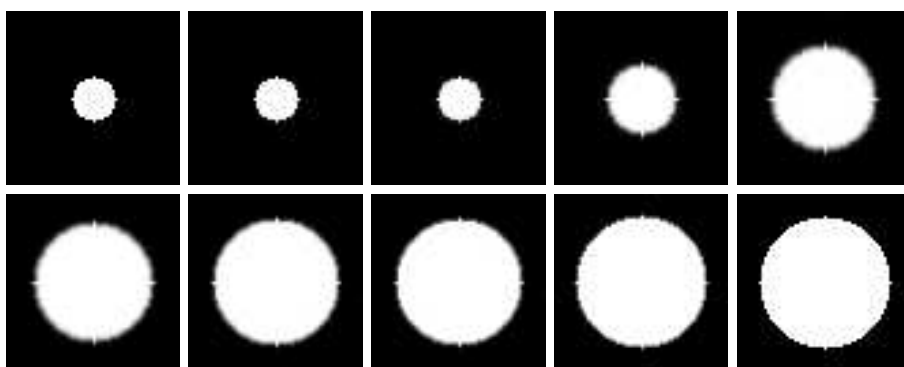
alors on définit R de la manière suivante

$$Rv = \sum_i \kappa_i v_i e_i$$

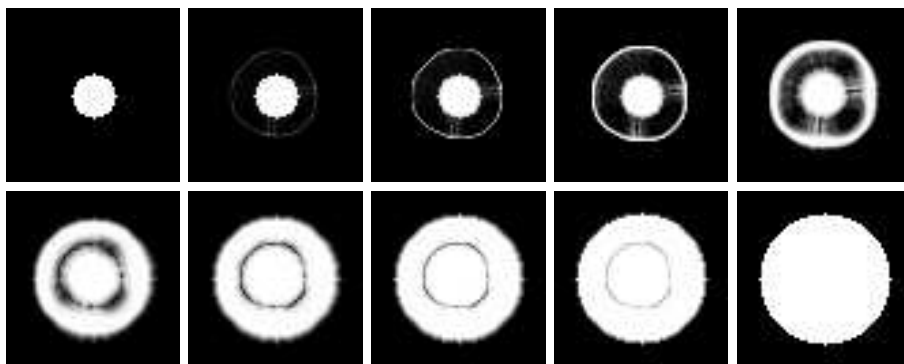
Autrement dit, on définit l’opérateur R à partir de ses vecteurs propres (la base orthonormale $\{e_i\}$) et de ses valeurs propres $\{\kappa_i\}$. L’espace de Hilbert \mathcal{V} est



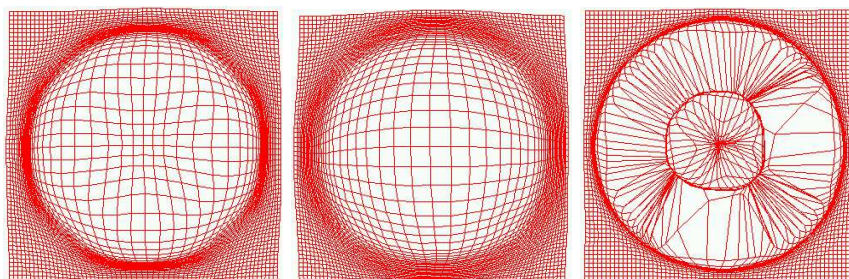
(a) $\sigma = 5$ pixels



(b) $\sigma = 10$ pixels



(c) $\sigma = 3$ pixels



(d) Déformation 3.4(a)

(e) Déformation 3.4(b)

(f) Déformation 3.4(c)

FIG. 3.4 – Influence de la taille du noyau gaussien régularisant sur la mise en correspondance du petit disque et du grand disque (taille de l'image 64 pixels)

alors l'ensemble des fonctions $v = \sum_i v_i e_i$ telles que

$$\|v\|_{\mathcal{V}}^2 = \sum_i \kappa_i^2 v_i^2 < \infty$$

L'énergie s'écrit alors :

$$E(\mathbf{J}, \mathbf{v}) = \int_0^1 \sum_i \kappa_i^2 \mathbf{v}_i(t)^2 dt + \int_0^1 \int_{\Omega} \left| \frac{\partial \mathbf{J}}{\partial t}(t, x) + \sum_i \mathbf{v}_i \langle \nabla \mathbf{J}(t, x), e_i(x) \rangle \right|^2 dx dt$$

Approche multi-résolution

Dans le cadre du calcul du flot optique (qui est, on le rappelle, l'analogie de notre problème mais dans un cadre de *petites* déformations), on s'est rapidement aperçu que la multi-résolution était un outil efficace (cf. [51, 2, 37] par exemple). Les ondelettes, de par leur structure multi-échelle, s'imposent alors naturellement comme un cadre d'analyse de la déformation [10, 9]. Mais, contrairement à ce qui a été fait auparavant, on va ici imposer des conditions de régularité sur la déformation par l'intermédiaire de l'opérateur R en tirant parti des résultats liant les coefficients de la décomposition en ondelettes et la régularité de la fonction analysée.

On décompose donc chaque composante du champ de déformation v dans une base d'ondelettes dyadiques 2D $\{\psi_{sk}\}$. Dans un but de simplification, on notera¹

$$v = \sum_{s,k} v_{sk} \psi_{sk}$$

où $s \in \mathbb{N}$ représente la résolution et $k = (k^1, k^2) \in \mathbb{Z}^2$ la position, c'est-à-dire que :

$$\psi_{sk}(x) = 2^{2s} \psi_{00}(2^s x - k)$$

Les v_{sk} sont des vecteurs de dimension 2. On peut retrouver cette méthode dans [3] mais dans un cadre différent de la mise en correspondance géodésique. On définit donc un opérateur R à partir de cette base d'ondelettes

$$Rv = \sum_{s,k} \kappa_{sk} v_{sk} \psi_{sk}$$

Mais on désire que la norme $\|R \cdot\|_{L^2}$ soit peu affecté par les translations. C'est-à-dire que si l'on note v^τ pour $v \in \mathcal{V}$ et $\tau \in \mathbb{R}^2$, l'application $x \rightarrow v(x - \tau)$, cela signifie que l'on souhaite avoir $\|Rv\|_{L^2} \simeq \|Rv^\tau\|_{L^2}$ pour tout $v \in \mathcal{V}$ et $\tau \in \mathbb{R}^2$. En particulier, on peut avoir pour tout s, k, k' , $\|R\psi_{sk}\|_{L^2} = \|R\psi_{sk'}\|_{L^2}$ en ne faisant pas dépendre les valeurs propres κ_{sk} de l'opérateur R de l'indice de position k , c'est-à-dire :

$$\forall k, \kappa_{sk} = \kappa_s$$

C'est le cas de figure dans lequel on se placera par la suite.

¹Par souci de clarté, on omettra volontairement le fait qu'il y ait en vérité trois types d'ondelettes en 2D.

1. Liens entre valeurs propres et régularité

Pour garantir une certaine régularité du champ de déformation v , on veut d'un point de vue qualitatif que les poids κ_s soient plus importants aux résolutions les plus fines qu'aux résolutions les plus grossières. On peut même se donner des critères quantitatifs sur la vitesse de croissance des poids κ_s en fonction de la résolution s pour imposer une régularité au champ v .

On rappelle d'abord la définition suivante :

Définition 3.1. Une fonction f est Lipschitz- α avec $\alpha = n + r$ ($n \in \mathbb{N}, 0 \leq r < 1$) s'il existe une constante C et un polynôme P de degré n tels que

$$\forall x, y, |f(x) - P(y)| < C|x - y|^\alpha$$

On peut alors énoncer le théorème suivant [34, 41, 40].

Théorème 3.2. Soit ψ_{sk} une base d'ondelettes 2D avec N moments nuls. Alors une fonction $f = \sum_{s,k} f_{sk} \psi_{sk}$, où $f_{sk} = \langle f, \psi_{sk} \rangle$, est Lipschitz- α avec $\alpha \leq N$ si et seulement si il existe une constante C telle que

$$|f_{sk}| < C2^{-2s(\alpha + \frac{1}{2})} \quad (3.3)$$

Par la suite, on supposera que la base d'ondelettes aura toujours un nombre N de moments nuls suffisant et que l'ondelette de base est à support compact (ondelettes de Daubechies par exemple).

Les deux propositions qui suivent donnent des relations entre les valeurs propres de l'opérateur R et la régularité des éléments de \mathcal{V} .

Proposition 3.3. Si les valeurs propres κ_s de l'opérateur R sont en $\mathcal{O}(2^{2s\beta})$ en fonction de la résolution s , alors toute fonction v à support compact et Lipschitz- α pour $\alpha > \beta$, appartient à l'espace de Hilbert \mathcal{V} défini par la norme $\|R\|_{L^2}$.

Démonstration. Soit v une fonction à support compact et Lipschitz- α pour $\alpha > \beta$. Alors, d'après Thm. 3.2, il existe une constante C telle que

$$|v_{sk}| < C2^{-2s(\alpha + \frac{1}{2})}$$

De plus v est à support compact donc on peut supposer qu'il existe des réels x_0, y_0, x_1, y_1 tels que :

$$\text{Supp}(v) \subset [x_0; y_0] \times [x_1; y_1]$$

Si l'on suppose de plus que

$$\text{Supp}(\psi_{00}) \subset [u_0; v_0] \times [u_1; v_1]$$

alors

$$\text{Supp}(\psi_{sk}) \subset [2^{-s}(u_0 + k^1); 2^{-s}(v_0 + k^2)] \times [2^{-s}(u_1 + k^1); 2^{-s}(v_1 + k^2)]$$

Donc les indices $k = (k^1, k^2)$ tels que v_{sk} soit non nul vérifient les inégalités :

$$\begin{aligned} 2^s x_0 + u_0 &\leq k^1 \leq 2^s x_1 + u_1 \\ 2^s y_0 + v_0 &\leq k^2 \leq 2^s y_1 + v_1 \end{aligned}$$

Le nombre N_s d'indices k tels que v_{sk} est non nul à la résolution s est donc majoré par $(2^s(x_1 - x_0) + (u_1 - u_0) + 1)(2^s(y_1 - y_0) + (v_1 - v_0) + 1)$. Donc en prenant $A = ((x_1 - x_0) + (u_1 - u_0) + 1)((y_1 - y_0) + (v_1 - v_0) + 1)$, on est assuré que pour tout $s \geq 0$, $N_s \leq A2^{2s}$. De plus, il existe une constante B indépendante de s telle que $\kappa_s \leq B2^{2s\beta}$. Donc, à une résolution s fixée

$$a_s = \kappa_s^2 \sum_k |v_{sk}|^2 \leq AB^2 C 2^{2s} 2^{4s\beta} 2^{-4s(\alpha + \frac{1}{2})} = \mathcal{O}(2^{4s(\beta - \alpha)})$$

La série des a_s converge et par conséquent v appartient à \mathcal{V} . \square

Proposition 3.4. *Si les valeurs propres κ_s de l'opérateur R sont en $\mathcal{O}(2^{2s\beta})$ en fonction de la résolution s , alors toute fonction v de \mathcal{V} est Lipschitz- α pour tout $\alpha < \beta - \frac{1}{2}$.*

Démonstration. Il existe à nouveau une constante B indépendante de s telle que $\kappa_s \leq B2^{2s\beta}$. On note $b_s = \sum_k |v_{sk}|^2$ à une résolution s donnée. Le fait que v appartient à \mathcal{V} se traduit par la relation

$$\sum_{s,k} \kappa_s^2 |v_{sk}|^2 \leq B^2 \sum_s b_s 2^{4s\beta} < \infty$$

Ceci signifie que la série entière $\sum_{s \geq 0} b_s z^s$ a un rayon de convergence au moins égal à $2^{4\beta}$. Et donc

$$\limsup (b_s)^{\frac{1}{s}} \leq 2^{-4\beta}$$

Et par conséquent, pour tout $\alpha < \beta - \frac{1}{2}$, il existe une constante C telle que pour tout s

$$b_s \leq C 2^{-4s(\alpha + \frac{1}{2})}$$

et comme $|v_{sk}| \leq b_s$ pour tout k , on obtient bien la relation (3.3) qui nous garantit que v est Lipschitz- α . \square

Les propositions 3.3 et 3.4 nous permettent donc de faire le lien entre la régularité souhaitée de la déformation et l'opérateur de régularisation à adopter.

2. Algorithme multi-résolution

L'intérêt de cette approche est que l'on peut calculer la déformation minimisant l'énergie à des résolutions différentes. Pour cela, il suffit de décomposer la déformation sur une partie seulement de la base d'ondelettes. Concrètement cela revient à ne prendre en compte les coefficients d'ondelettes v_{sk} du champ de déformation v que pour des résolutions $s \leq S$ où S est la résolution souhaitée.

On peut donc alors mettre en place une stratégie qui vise à estimer dans un premier temps une déformation grossière qu'on raffinerait au fur et à mesure [5]. C'est-à-dire qu'on itère les minimisations de l'énergie en partant de la résolution d'analyse en ondelettes la plus grossière et en allant jusqu'à la résolution la plus fine. A chaque minimisation, on prend comme initialisation de la déformation à la résolution s le résultat de la minimisation à la résolution précédente $s - 1$ (cf. Alg. 1).

On constate que le résultat final a une énergie plus basse que si l'on minimisait directement sur toutes les résolutions à la fois. Ce phénomène est visible en comparant les énergies finales dans la figures 3.9 et, de manière plus visuelle, en comparant l'évolution de l'image avec (fig. 3.5(c)) ou sans multi-résolution (fig. 3.5(b)). Les paramètres sont pourtant dans les deux cas identiques, mais dans le premier cas (fig. 3.5(b)), il semble que l'algorithme se retrouve bloqué dans un minimum local tandis que dans le second cas (fig. 3.5(b)), l'évolution est plus souple. On s'aperçoit aussi en comparant les déformations (fig. 3.7) que dans le cas sans multi-résolution (fig. 3.7(a)), les déformations semblent plus locales que dans le cas avec multi-résolution (fig. 3.7(b)). La figure 3.8 le prouve de manière flagrante : dans le cas sans multi-résolution, les grandes déformations au niveau de la bouche ou du nez ne sont pas du tout retrouvées au contraire du cas avec multi-résolution.

Cela est dû au fait que dans le cas multi-résolution, l'algorithme a pu trouver les grandes déformations en travaillant d'abord aux résolutions les plus grossières puis en affinant l'évaluation des déformations aux échelles les plus fines comme le montre la figure 3.11 sur un nouvel exemple. On peut aussi se rendre compte de ce phénomène en observant les jacobiens des déformations retrouvées aux différentes échelles dans la figure 3.15, où l'on s'aperçoit que la déformation est d'autant plus détaillée que la résolution est fine.

On peut également mettre en évidence l'influence du facteur de croissance β des valeurs propres κ_s de l'opérateur R ($\kappa_s = 2^{2s\beta}$) en observant les différentes séries d'images à des échelles différentes et pour des β différents (cf. fig. 3.11, 3.10, 3.12, 3.13). Mais l'évolution du jacobien de la déformation en fonction des échelles et du facteur β est sans doute encore plus parlante (cf. fig. 3.15, 3.14, 3.16, 3.17). On voit bien qu'un β trop fort pénalise les petites déformations et qu'inversement un β trop faible donne des déformations trop irrégulières.

Algorithm 1 Mise en correspondance de deux images I_0 et I_1 en multi-résolution

Initialiser les déformations $\mathbf{v}(t) = 0$

Initialiser les images $J(t) = (1 - t)I_0 + tI_1$

Initialiser l'échelle $S = 0$

repeat

repeat

 Descente de gradient en la déformation \Rightarrow calcul des \mathbf{v}_{sk} pour $s \leq S$

 Descente de gradient en l'image

until $\Delta E < \epsilon$

$S \leftarrow S + 1$

until $S > S_{max}$

Détermination d'un opérateur de régularisation approprié par apprentissage

Les résultats des mises en correspondance sont déjà d'une qualité très satisfaisante. Néanmoins, on souhaite déterminer un opérateur de régularisation qui

soit encore plus adapté à certains types de problèmes. En effet, dans certaines applications, les déformations peuvent avoir une forme caractéristique. On va donc chercher à déterminer les valeurs propres de l'opérateur R par apprentissage. Pour ce faire, on va étudier le comportement statistique des coefficients v_i dans la décomposition $v = \sum v_i e_i$. Les v_i sont des vecteurs de dimension 2. On calcule donc leurs matrices de covariance Σ_i par apprentissage et en supposant que les v_i suivent des lois normales, on définit l'opérateur de régularisation R de manière à ce que la norme $\|\cdot\|_{\mathcal{V}}$ définisse une "distance" d'un élément v de \mathcal{V} à la loi gaussienne définie par les matrices de covariance Σ_i :

$$v = \sum_i v_i e_i \Rightarrow Rv = \sum_i \Sigma_i^{-\frac{1}{2}} v_i e_i \Rightarrow \|v\|_{\mathcal{V}}^2 = \sum_i v_i^T \Sigma_i^{-1} v_i$$

Il est important de noter que si les v_i sont des vecteurs gaussiens alors la norme de v définie plus haut est presque sûrement infinie. On se tire de ce mauvais pas en considérant dans la suite de l'exposé que l'on observe qu'un nombre fini de composantes v_i , ce qui est le cas dans la pratique. En effet, après discrétisation du problème, la base (e_i) de décomposition de la déformation v est de cardinal fini.

1. Analyse en ondelettes puis en Fourier

Dans cette partie, nous considérerons le champ de déplacement v comme étant discrétisé sur une grille régulière. On le décompose sur une base d'ondelettes discrètes – c'est à dire qu'on effectue une FWT :

$$v(x) = \sum_{s,k,n} c_{s,k}^n \psi_{s,k}^n(x)$$

où s et k représentent l'échelle et la position du coefficient d'ondelettes et $n = 0, 1, 2$ indexe les trois types de fonctions d'ondelettes entrant en jeu dans la décomposition en 2D.

On effectue alors une FFT sur chaque échelle de la transformée en ondelettes de v , c'est-à-dire que l'on définit les coefficients suivants :

$$v_{s,q}^n = \sum_k c_{s,k}^n e_q(-k)$$

où e_q est la base des exponentielles complexes. On a par transformation inverse :

$$c_{s,k}^n = \sum_q v_{s,q}^n e_k(q)$$

On considère alors les variables $v_{s,q}^n$ comme des variables aléatoires gaussiennes que l'on suppose non corrélées. Si on note alors :

$$\hat{\psi}_{s,q}^n(x) = \sum_k \psi_{s,k}^n(x) e_q(k)$$

on peut écrire :

$$v(x) = \sum_{s,q,n} v_{s,q}^n \hat{\psi}_{s,q}^n(x)$$

On suppose alors une stationnarité du champ $c_s = (c_s^n)_{n=0,1,2}$ à chaque échelle s . On s'intéresse donc à la structure de covariance suivante :

$$E [c_{s,k} c_{s,k+k'}^T] = \sum_{q,q'} E[v_{s,q} v_{s,q'}^T] e_q(k) e_{q'}(k+k')$$

où $v_{s,q}$ est le vecteur $(v_{s,q}^n)_{n=0,1,2}$. Les variables $v_{s,q}$ étant non corrélées, on a $E[v_{s,q} v_{s,q'}^T] = 0$ sauf pour $q = q'$ et $q = -q'$ (le champ c_s étant réel, on a la relation : $v_{s,-q} = v_{s,q}^*$). Il s'ensuit :

$$E [c_{s,k} c_{s,k+k'}^T] = \sum_q E[v_{s,q} v_{s,q}^T] e_q(2k+k') + \sum_q E[v_{s,q} v_{s,q}^{*T}] e_q(-k')$$

Le champ c_s étant stationnaire, l'espérance $E [c_{s,k} c_{s,k+k'}^T]$ ne doit donc dépendre que de k' . Ainsi, si l'on note $v_{s,q}^n = \alpha_{s,q}^n + i\beta_{s,q}^n$, on a pour tout q, n, m :

$$E[v_{s,q}^n v_{s,q}^m] = 0 \Rightarrow \begin{cases} E[\alpha_{s,q}^n \alpha_{s,q}^m] = E[\beta_{s,q}^n \beta_{s,q}^m] \\ E[\alpha_{s,q}^n \beta_{s,q}^m] + E[\alpha_{s,q}^m \beta_{s,q}^n] = 0 \end{cases}$$

$$Re(E[v_{s,q}^n v_{s,q}^{m*}]) = E[\alpha_{s,q}^n \alpha_{s,q}^m] + E[\beta_{s,q}^n \beta_{s,q}^m]$$

- On décide donc de modéliser le champ aléatoire v de la manière suivante :
- les variables aléatoires $\alpha_{s,q}$ et $\beta_{s,q}$ sont indépendantes et suivent des lois gaussiennes de moyennes nulles et de matrices de covariance $\Sigma_{s,q} = Re(E[v_{s,q} v_{s,q}^{*T}]/2)$
 - par symétrie, $\alpha_{s,-q} = \alpha_{s,q}$ et $\beta_{s,-q} = -\beta_{s,q}$.
 - on obtient les différentes échelles de la transformée en ondelettes de v par des transformées de Fourier inverses i.e. $c_s = \mathcal{F}^{-1}(v_s)$
 - et on obtient le champ de vitesse v par une transformée en ondelettes inverse i.e. $v = \mathcal{W}^{-1}(c)$

Les matrices de covariance $\Sigma_{s,q}$ déterminent alors complètement la structure du champ de déplacement aléatoire v .

2. Introduction d'une norme ad hoc

On introduit une nouvelle norme N_G (cf. 2.4.1) sur les champs de déplacement de la manière suivante :

$$N_G(v) = \sum_{s,q} v_{s,q}^{*T} \Sigma_{s,q}^{-1} v_{s,q}$$

où $v_{s,q} = (v_{s,q}^n)_{0 \leq n \leq 2}$.

Si l'on veut garder le formalisme des opérateurs alors il suffit de définir l'opérateur R de la manière suivante :

$$R = \Sigma^{-\frac{1}{2}} \mathcal{F} \mathcal{W}$$

3. Détermination des lois gaussiennes par apprentissage

Nous allons déterminer les matrices de covariance $\Sigma_{s,q}$ par apprentissage. On synthétise alors N couples de MNE pour chacun desquels on connaît le champ de déplacement v sous l'hypothèse des petites déformations. On calcule alors pour chaque couple les vecteurs $v_{s,q}$ et on calcule les matrices de covariance $\Sigma_{s,q}$ de ces vecteurs à l'aide des N estimations.

4. Résultats

Afin de comparer la mise en correspondance géodésique avec ou sans apprentissage, on va comparer au MNE initial le MNE initial déformé par le difféomorphisme retrouvé dans chacun des cas (cf. fig. 3.18). Dans le cas sans apprentissage, l'algorithme a généré des déformations là où de nouveaux bâtiments apparaissent (dans le coin supérieur droit notamment), ce qui n'est pas le cas dans le cas avec apprentissage. Il semblerait donc que l'apprentissage permette de générer des déformations un peu plus "réalistes" sur cet exemple.

3.2 Algorithme

3.2.1 Version discrétisée de l'énergie

On ne peut pas utiliser l'expression (3.1) pour la minimisation de l'énergie. En effet, la discrétisation directe du terme de "flot optique" dans (3.1) conduit à une instabilité numérique.

On donne donc une version discrétisée en temps de l'énergie qui nous servira par la suite à expliciter la minimisation numérique de l'énergie :

$$E(\mathbf{J}, \mathbf{v}) = \sum_{t=0}^{NT-1} \left[\lambda \|\mathbf{v}(t, \cdot)\|_{\mathcal{V}}^2 + \int_{\Omega} |\mathbf{J}(t+1, x + \mathbf{v}(x, t)) - \mathbf{J}(x, t)|^2 dx \right] \quad (3.4)$$

où NT est le nombre de pas de temps utilisés dans cette discrétisation.

On discrétise ensuite \mathbf{J} et \mathbf{v} sur une grille rectangulaire qu'on identifiera à \mathbb{Z}^2 . On considère donc \mathbf{J} et \mathbf{v} comme des applications de \mathbb{Z}^2 dans \mathbb{R} et \mathbb{R}^2 respectivement. A cet effet, on notera \mathcal{J} l'opérateur d'interpolation bilinéaire :

$$\begin{aligned} \mathcal{J} : \mathbb{R}^{\mathbb{Z}^2} \times \mathbb{R}^2 &\rightarrow \mathbb{R} \\ (J, (x, y)) &\rightarrow \begin{aligned} &(1 - r(x))(1 - r(y))J(\lfloor x \rfloor, \lfloor y \rfloor) \\ &+ r(x)(1 - r(y))J(\lfloor x \rfloor + 1, \lfloor y \rfloor) \\ &+ (1 - r(x))r(y)J(\lfloor x \rfloor, \lfloor y \rfloor + 1) \\ &+ r(x)r(y)J(\lfloor x \rfloor + 1, \lfloor y \rfloor + 1) \end{aligned} \end{aligned}$$

où $r(z)$ représente la partie fractionnaire du réel z et $\lfloor z \rfloor$ sa partie entière.

On minimisera cette énergie par des descentes de gradient alternées en \mathbf{J} et en \mathbf{v} . Reste alors à calculer les versions discrètes de ces gradients.

3.2.2 Calcul du gradient en la déformation

Le gradient dépend des variables que l'on prend pour définir la déformation (lissage gaussien ou transformée en ondelettes).

On posera par la suite :

$$f(t, x) = (\mathbf{J}(t+1, x + \mathbf{v}(t, x)) - \mathbf{J}(t, x)) \nabla \mathbf{J}(t+1, x + \mathbf{v}(t, x))$$

Cas gaussien

L'énergie discrétisée en temps s'écrit alors dans ce cas :

$$E(\mathbf{J}, \mathbf{w}) = \sum_{t=0}^{NT-1} \left[\lambda \int_{\Omega} |\mathbf{w}(t, x)|^2 dx + \int_{\Omega} |\mathbf{J}(t+1, x + N\mathbf{w}(t, x)) - \mathbf{J}(t, x)|^2 dx \right]$$

où N est le noyau gaussien et $\mathbf{v}(t) = N\mathbf{w}(t)$.

On peut alors calculer le gradient en \mathbf{w} de l'énergie :

$$\frac{1}{2} \nabla_{\mathbf{w}} E(t) = \lambda \mathbf{w} + Nf \quad (3.5)$$

Il s'agit là du gradient pour la version de l'énergie discrétisée en la seule variable temporelle. Dans le cas, où l'énergie est aussi discrétisée en la variable spatiale, l'énergie de mise en correspondance devient :

$$E(\mathbf{J}, \mathbf{v}) = \sum_{t=0}^{NT-1} \sum_{(x,y) \in \mathbb{Z}^2} [\lambda |\mathbf{w}(t, x, y)|^2 + |\mathfrak{J}(\mathbf{J}(t+1), (x, y) + \mathbf{v}(t, x, y)) - \mathbf{J}(t, x, y)|^2]$$

Cependant le calcul du gradient pour l'énergie discrétisée en espace suit immédiatement : il suffit pour cela d'utiliser une convolution discrète au lieu d'une convolution continue dans (3.5).

Cas des ondelettes

La version discrétisée en temps de l'énergie peut alors s'écrire :

$$E(\mathbf{J}, \mathbf{v}) = \sum_{t=0}^{NT-1} \left[\sum_{s,k} \kappa_s^2 |\mathbf{v}_{sk}(t)|^2 + \int_{\Omega} \left| \mathbf{J} \left(t+1, x + \sum_{s,k} \mathbf{v}_{sk}(t) \psi_{sk}(x) \right) - \mathbf{J}(t, x) \right|^2 dx \right]$$

On peut donc en tirer le gradient de l'énergie en les coefficients $\mathbf{v}_{sk}(t)$:

$$\frac{1}{2} \nabla_{\mathbf{v}_{sk}(t)} E = \kappa_s^2 \mathbf{v}_{sk}(t) + \langle f(t), \psi_{sk} \rangle \quad (3.6)$$

Il s'agit encore de la version du gradient pour l'énergie discrétisée en la seule variable temporelle. L'énergie discrétisée en temps et en espace s'écrit :

$$E(\mathbf{J}, \mathbf{v}) = \sum_{t=0}^{NT-1} \left[\sum_{s \geq 0, (k,l) \in \mathbb{Z}^2} \kappa_s^2 |\mathbf{v}_{sk}(t)|^2 + \sum_{(x,y) \in \mathbb{Z}^2} |\mathfrak{J}(\mathbf{J}(t+1), (x, y) + v(x, y)) - \mathbf{J}(t, x, y)|^2 \right]$$

avec $v(x, y) = \sum_{s \geq 0, (k,l) \in \mathbb{Z}^2} \mathbf{v}_{sk}(t) \psi_{sk}(x, y)$. Mais une fois encore, le calcul du gradient dans le cas où l'énergie est discrétisée en temps et en espace est immédiat : il suffit d'utiliser la transformée en ondelettes discrète dans (3.6).

3.2.3 Calcul du gradient en l'image

Le gradient de l'énergie en \mathbf{J} est indépendant de la norme en \mathbf{v} adoptée selon les applications :

$$\frac{1}{2} \nabla_{\mathbf{J}} E = A_{\mathbf{v}}^T A_{\mathbf{v}} \mathbf{J}$$

où on note $A_{\mathbf{v}}$ l'opérateur linéaire qui à \mathbf{J} associe $A_{\mathbf{v}} \mathbf{J}$ tel que :

$$(A_{\mathbf{v}} \mathbf{J})(t, x) = \mathbf{J}(t+1, x + \mathbf{v}(t, x)) - \mathbf{J}(t, x)$$

Mais ceci correspond au gradient en \mathbf{J} pour une version spatialement continue de l'énergie. Pour une version discrétisée en espace, il faut être plus soigneux dans le calcul du gradient et notamment prêter attention à la façon dont les valeurs des différentes variables sont interpolées en dehors de la grille discrète.

On note ainsi $\aleph_{\mathbf{v}}$ l'opérateur linéaire qui à une image discrète J associe l'image $\aleph_{\mathbf{v}} J$ interpolée sur la grille $G_{\mathbf{v}}$:

$$G_{\mathbf{v}} = \{(x, y) + v(x, y) | (x, y) \in \mathbb{Z}^2\}$$

c'est-à-dire que :

$$(\aleph_{\mathbf{v}} J)(x, y) = \mathcal{J}(J, (x, y) + v(x, y))$$

L'énergie peut alors s'écrire au moyen de cet opérateur :

$$E(\mathbf{J}, \mathbf{v}) = \sum_{t=0}^{NT-1} [\|\mathbf{v}(t)\|_{\mathbf{v}}^2 + \lambda \|\aleph_{\mathbf{v}(t)} \mathbf{J}(t+1) - \mathbf{J}(t)\|^2]$$

Et le gradient de l'énergie en l'image est :

$$\frac{1}{2\lambda} \nabla_{\mathbf{J}(t)} E = \mathbf{J}(t) - \aleph_{\mathbf{v}(t)} \mathbf{J}(t+1) + \aleph_{\mathbf{v}(t-1)}^T (\aleph_{\mathbf{v}(t-1)} \mathbf{J}(t) - \mathbf{J}(t-1))$$

Il faut maintenant définir explicitement l'opérateur $\aleph_{\mathbf{v}}^T$ pour pouvoir calculer le gradient de l'énergie en \mathbf{J} .

Pour cela, on se donne une position $(x_0, y_0) \in \mathbb{Z}^2$ et une image I telle que pour tout $(x, y) \in \mathbb{Z}^2$, $I(x, y) = \delta(x_0, x) \delta(y_0, y)$ où δ est le symbole de Kronecker.

Pour $(k, l) \in \mathbb{Z}^2$, on note $P_{(k,l)}$ le pavé de \mathbb{R}^2 défini comme suit :

$$P_{(k,l)} = \{(x, y) | k \leq x < k+1, l \leq y < l+1\}$$

On a alors pour toute image J :

$$\begin{aligned} (\aleph_{\mathbf{v}} I, J) &= \sum_{(x,y) \in \mathbb{Z}^2 | (x',y') \in P_{(x_0,y_0)}} (1-r(x'))(1-r(y')) J(x, y) \\ &+ \sum_{(x,y) \in \mathbb{Z}^2 | (x',y') \in P_{(x_0-1,y_0)}} r(x')(1-r(y')) J(x, y) \\ &+ \sum_{(x,y) \in \mathbb{Z}^2 | (x',y') \in P_{(x_0,y_0-1)}} (1-r(x'))r(y') J(x, y) \\ &+ \sum_{(x,y) \in \mathbb{Z}^2 | (x',y') \in P_{(x_0-1,y_0-1)}} r(x')r(y') J(x, y) \end{aligned}$$

en notant $(x', y') = (x, y) + v(x, y)$.

Or l'opérateur \aleph_v^T est défini par la relation $(\aleph_v I, J) = (I, \aleph_v^T J)$ pour tout I, J . Donc, dans notre cas particulier :

$$(\aleph_v I, J) = (I, \aleph_v^T J) = (\aleph_v^T J)(x_0, y_0)$$

Dans la pratique, les images ne sont pas définies sur \mathbf{Z}^2 mais sur un pavé $\{0, \dots, N_x - 1\} \times \{0, \dots, N_y - 1\}$ de \mathbf{Z}^2 où N_x et N_y sont respectivement le nombre de colonnes et de lignes de l'image. On prolonge une image J définie sur $\{0, \dots, N_x - 1\} \times \{0, \dots, N_y - 1\}$ en une image \tilde{J} définie sur \mathbf{Z}^2 en posant pour $(k, l) \in \mathbf{Z}^2$

$$\tilde{J}(k, l) = J(\min(\max(k, 0), N_x - 1), \min(\max(l, 0), N_y - 1))$$

En fait, on ne prolonge pas l'image sur \mathbf{Z}^2 tout entier mais sur un pavé de \mathbf{Z}^2 en dehors duquel on est sûr que la déformation est nulle. Si on note T la taille du noyau de convolution discret ou de l'ondelette de base discrète, il suffit donc de prolonger les images sur $\{-T, \dots, N_x - 1 + T\} \times \{-T, \dots, N_y - 1 + T\}$.

3.3 Conclusion

Dans ce chapitre, on a pu mettre au point un algorithme de mise en correspondance d'images ou de MNE (de surfaces si l'on veut être le plus général possible) qui cumule plusieurs avantages :

- le cadre théorique des grandes déformations,
- le choix d'un opérateur de régularisation adapté (décomposition sur une base d'ondelettes),
- la mise en correspondance multi-résolution ("coarse to fine").

Les expériences montrent en effet que chacun de ces trois aspects génère un véritable gain dans la qualité de la mise en correspondance. Notre définition de l'opérateur de régularisation nous permet aussi de contrôler la régularité de la déformation. On fait aussi remarquer que l'algorithme se comporte correctement même quand on n'impose qu'une faible régularité à la déformation : c'est ce qui nous permet de retrouver des déformations d'une précision très satisfaisante.

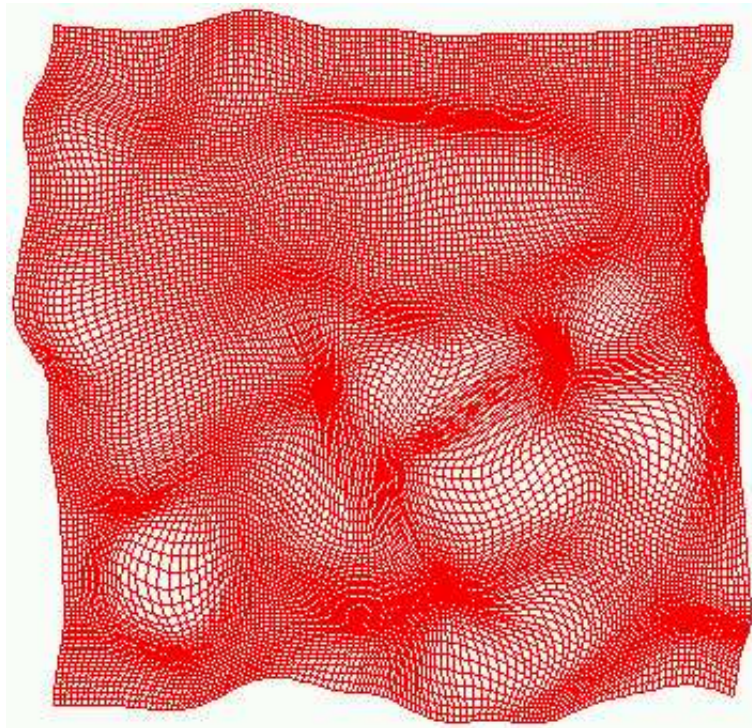
(a) Mise en correspondance avec régularisation gaussienne (taille du noyau $\sigma = 5$ pixels)

(b) Mise en correspondance par ondelettes sans multi-résolution

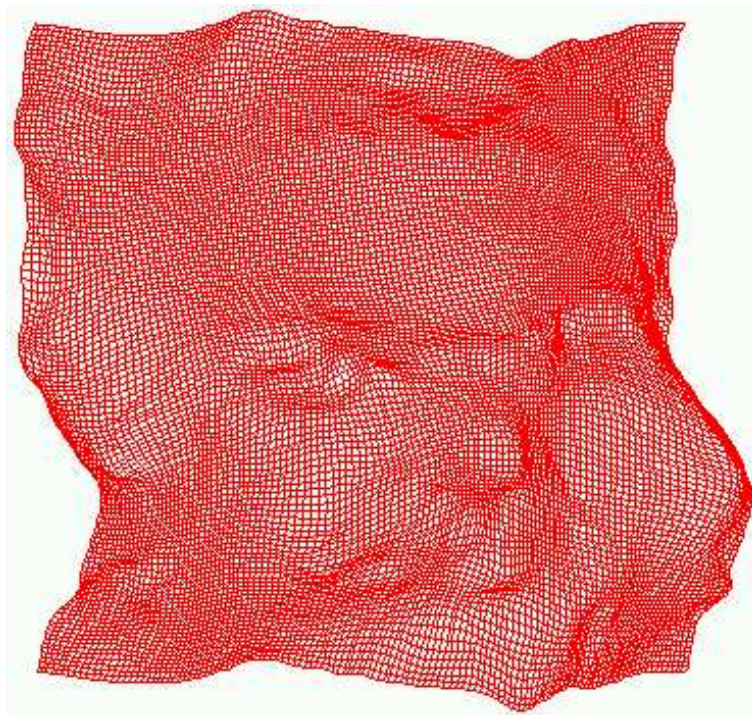


(c) Mise en correspondance par ondelettes avec multi-résolution

FIG. 3.5 – Comparaison des différents types de mise en correspondance. Les images entre la première et la dernière de chaque série sont calculées par l'algorithme

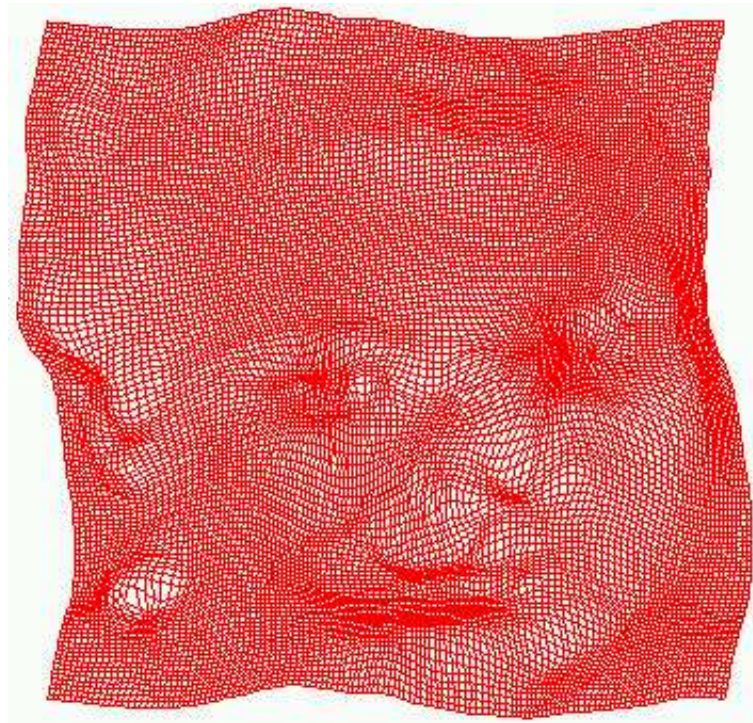


(a) Déformation gaussienne

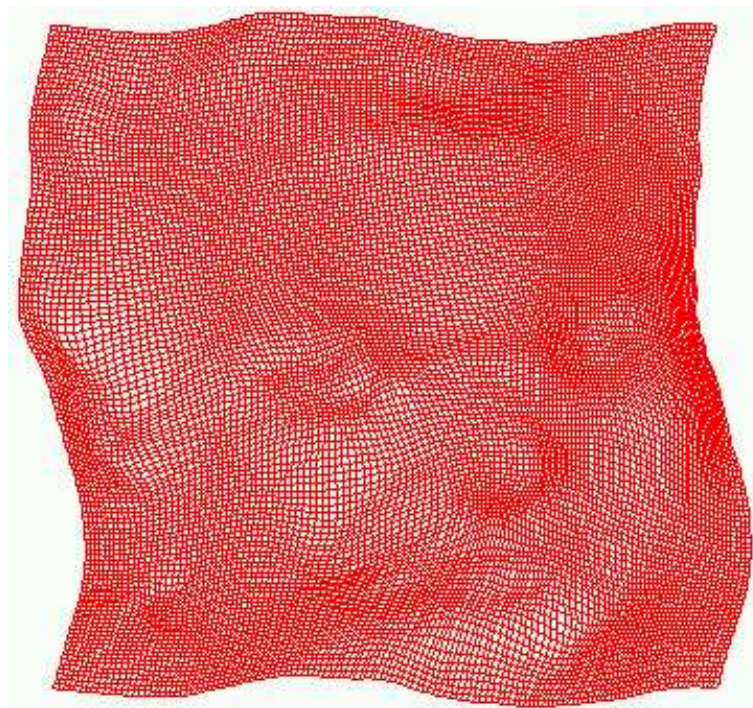


(b) Déformation ondelettes avec multi-résolution (petites déformations)

FIG. 3.6 – Comparaison des déformations pour les différents types de mise en correspondance



(a) Déformation ondelettes sans multi-résolution



(b) Déformation ondelettes avec multi-résolution

FIG. 3.7 – Comparaison des déformations pour les différents types de mise en correspondance (suite)



(a) Image initiale et image finale



(b) Mise en correspondance gaussienne



(c) Mise en correspondance ondelettes sans multi-résolution

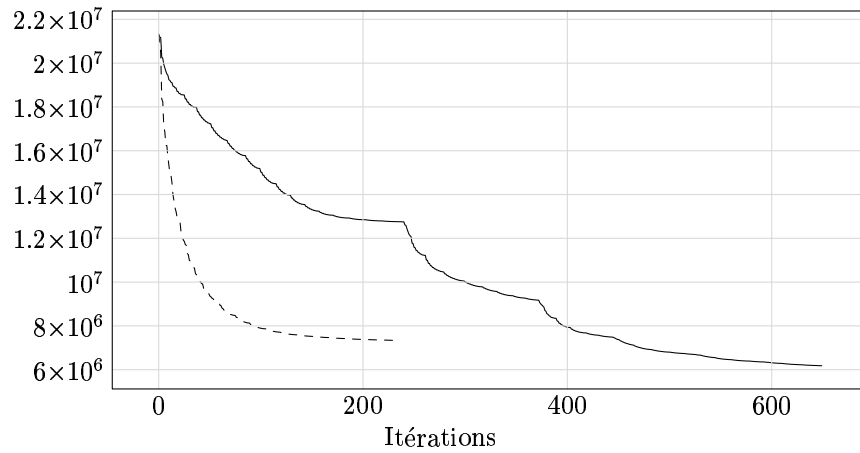


(d) Mise en correspondance ondelettes avec multi-résolution (petites déformations)

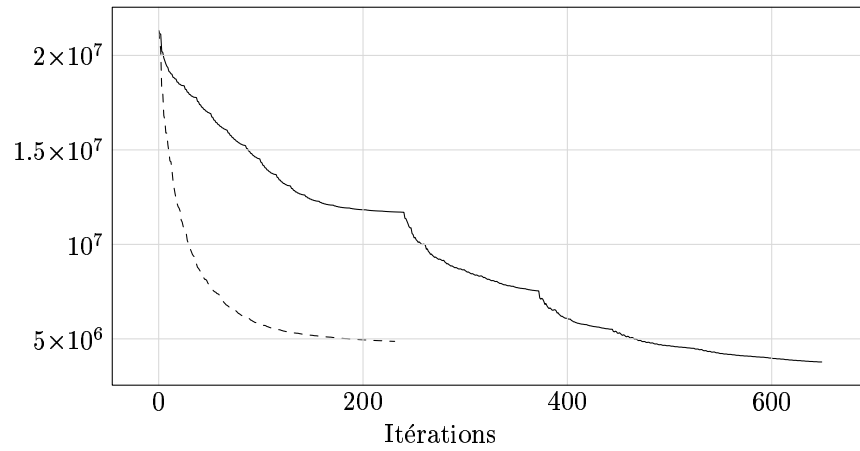


(e) Mise en correspondance ondelettes avec multi-résolution

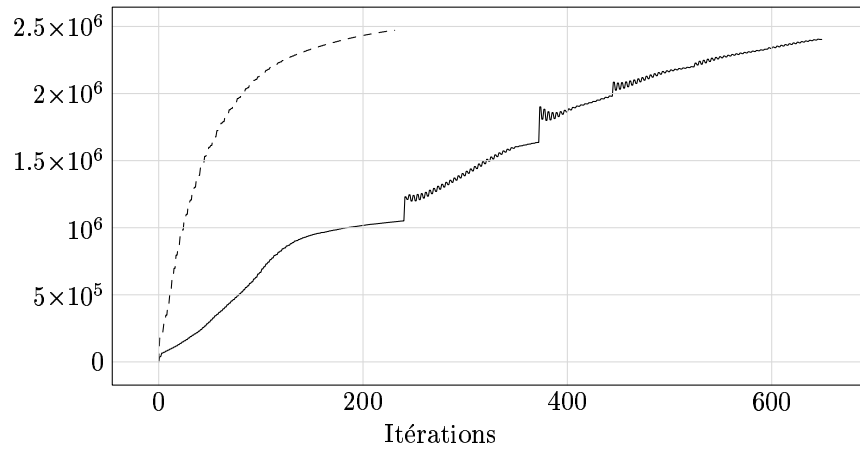
FIG. 3.8 – Dans chaque paire : image finale transformée par la déformation inverse (à comparer à l'image initiale) et image initiale transformée par la déformation directe (à comparer à l'image finale)



(a) Energie totale



(b) Energie de flot



(c) Energie de régularisation

FIG. 3.9 – Variation des énergies au cours de l'algorithme à base d'ondelettes avec multi-résolution (continu) et sans multi-résolution (pointillés)



(a) Echelle 1



(b) Echelle 3



(c) Echelle 8

FIG. 3.10 – Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opérateur de régularisation $\beta = 0$



(a) Echelle 1



(b) Echelle 3



(c) Echelle 8

FIG. 3.11 – Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opérateur de régularisation $\beta = 0.5$



(a) Echelle 1



(b) Echelle 3



(c) Echelle 8

FIG. 3.12 – Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opérateur de régularisation $\beta = 1$



(a) Echelle 1



(b) Echelle 3



(c) Echelle 8

FIG. 3.13 – Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opérateur de régularisation $\beta = 1.5$

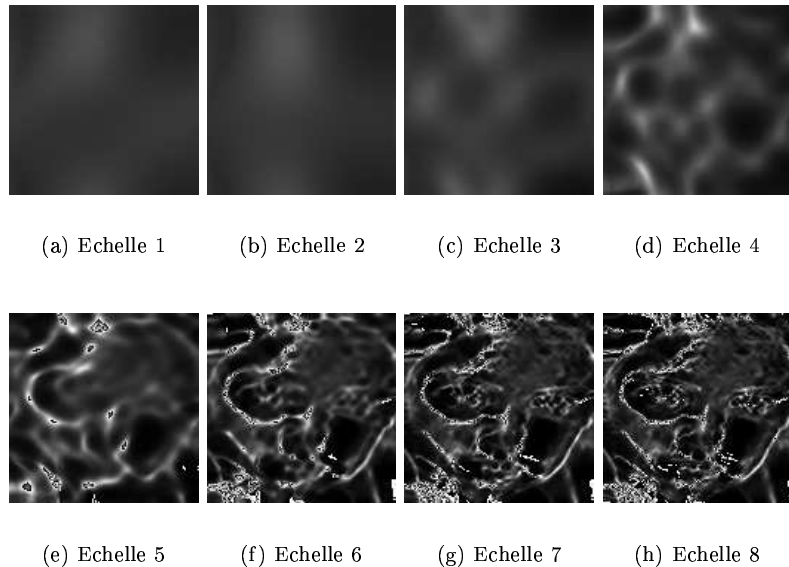


FIG. 3.14 – Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 0$)

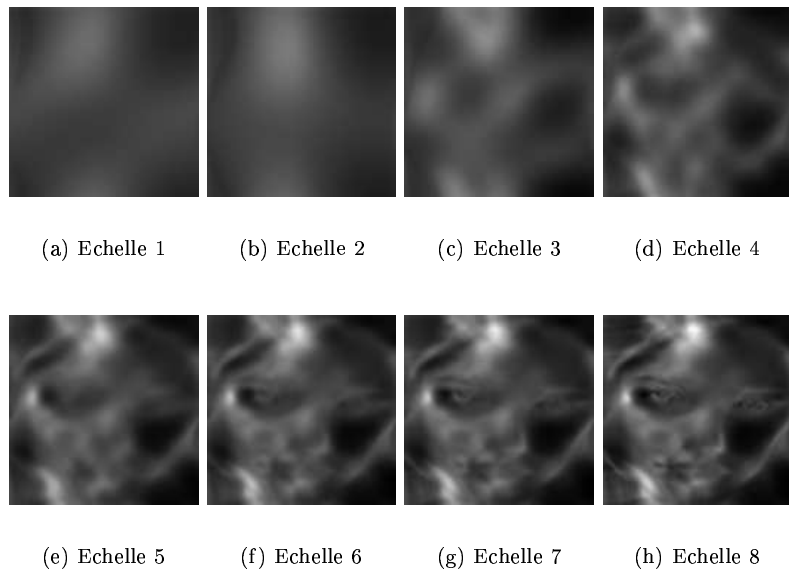


FIG. 3.15 – Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 0.5$)

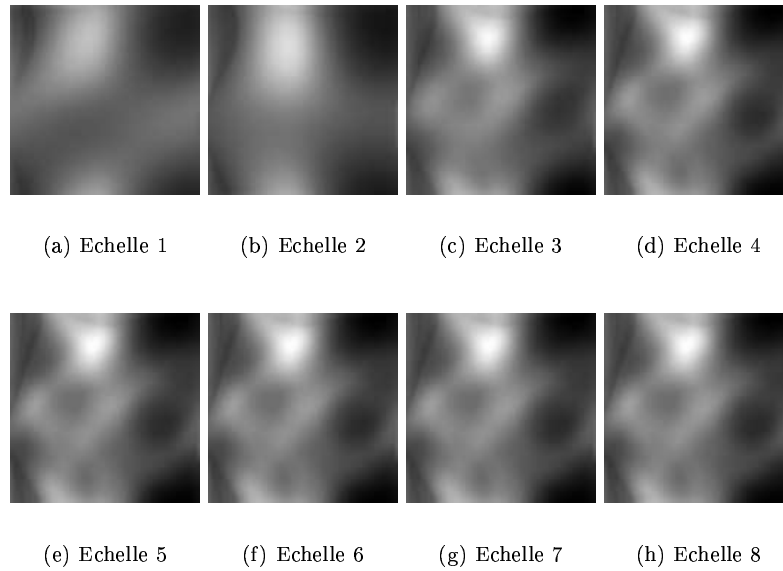


FIG. 3.16 – Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 1$)

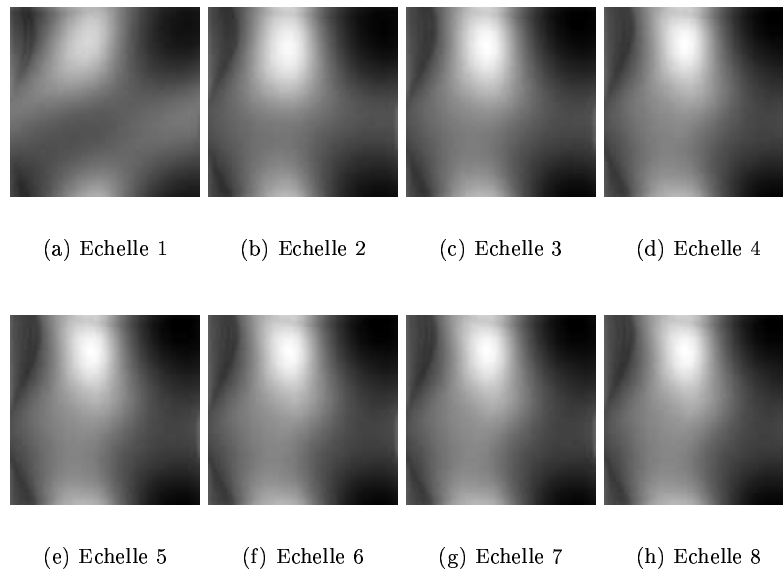
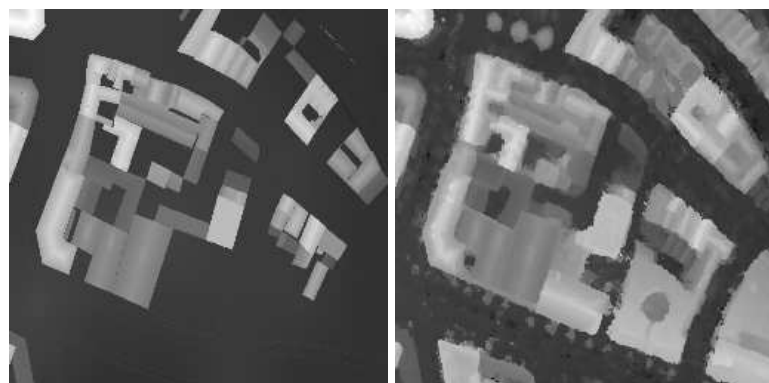
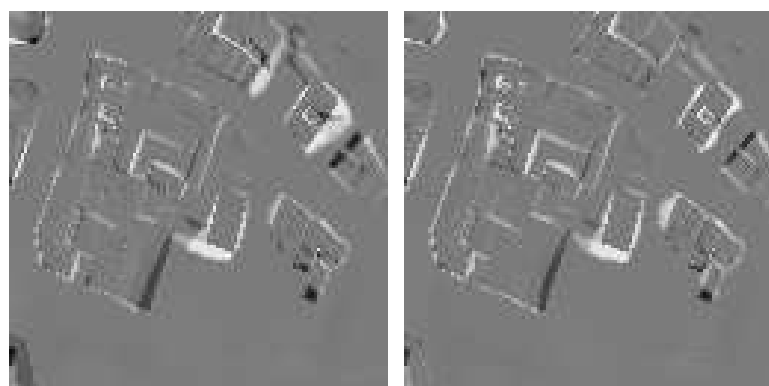


FIG. 3.17 – Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 1.5$)



(a) MNE de départ

(b) MNE d'arrivée



(c)

(d)

FIG. 3.18 – Comparaison de l'image initiale et de l'image initiale déformée sans apprentissage (fig. 3.18(c)) et avec apprentissage (fig. 3.18(d)).

Chapitre 4

Mise en correspondance de primitives

On utilisera dans ce chapitre la théorie des noyaux reproduisants introduite par Aronszajn [4] dans le contexte de l'interpolation par spline (cf. [57] pour des exemples d'application) afin d'interpoler des déformations entre primitives. Nous adapterons cette théorie afin de prendre en compte des éléments de premier ordre des primitives i.e. des tangentes ou des normales en plus des points. On verra que la prise en compte de ces caractéristiques de premier ordre permet à l'algorithme de mise en correspondance de gagner en robustesse tant dans la mise en correspondance des primitives (cf. chap. 5) que dans l'estimation de la déformation. Dans ce chapitre nous ne traiterons pour l'instant que le problème de l'estimation de la déformation, la mise en correspondance étant supposée connue.

4.1 Modélisation de primitives

On définit de manière générale une primitive par un certain nombre de points associés ou non à des vecteurs (tangentes, normales). Des exemples concrets seront donnés dans les sections 5.5 et 6.6. La figure 4.1 donne un exemple de la définition d'une courbe par des points interpolant associés ou non à des vecteurs (tangentes ou normales).

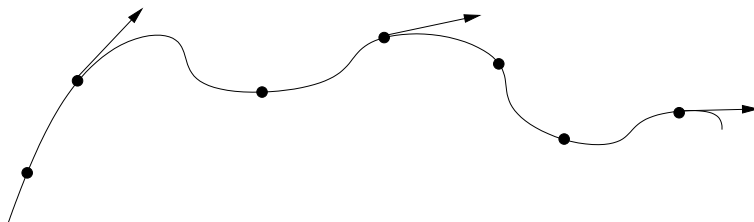


FIG. 4.1 – Une courbe définie par des points et des tangentes

4.2 Interpolation par splines

Dans la suite de notre travail, nous aurons besoin de définir des déformations capables de mettre en correspondance différentes sortes de primitives. Comme nous avons défini nos objets à l'aide de points (cf. section 4.1, l'approche par splines semble naturelle. Cependant, comme nous faisons aussi intervenir des vecteurs (tangentes ou normales), nous devons définir ce que nous appellerons des splines de premier ordre (cf. 4.2.3).

4.2.1 Notations

Soit \mathcal{H} un espace de Hilbert inclus dans $L^2(\Omega, \mathbb{R}^d)$ – Ω étant un ouvert de \mathbb{R}^d – doté d'un produit scalaire $\langle \cdot, \cdot \rangle$ et d'une norme $\|\cdot\|$. On définit :

$$\forall e \in \mathcal{H}^d, \alpha \in \mathbb{R}^d, \quad e \cdot \alpha = \sum_{i=1}^d e^i \alpha^i \in \mathcal{H} \quad (4.1)$$

$$\forall e \in \mathcal{H}^d, h \in \mathcal{H}, \quad \langle e, h \rangle = (\langle e^i, h \rangle)_{i=1, \dots, d} \in \mathbb{R}^d \quad (4.2)$$

$$\forall e \in \mathcal{H}^d, e' \in \mathcal{H}^{d'}, \quad \langle e, e' \rangle = (\langle e^i, e'^j \rangle) \begin{cases} 1 \leq i \leq d \\ 1 \leq j \leq d' \end{cases} \in \mathcal{M}_{d, d'}(\mathbb{R}) \quad (4.3)$$

4.2.2 Problème générique

En premier lieu, examinons un problème plus général, à savoir la minimisation d'une énergie du type suivant :

$$E(h) = \|h\|^2 + \sum_{k=1}^N \xi_k (\langle e_k, h \rangle - c_k) \quad (4.4)$$

où h appartient à \mathcal{H} , les e_k à \mathcal{H}^{d_k} , les c_k à \mathbb{R}^{d_k} et les ξ_k sont des formes quadratiques de \mathbb{R}^{d_k} .

Proposition 4.1. *L'élément \hat{h} de \mathcal{H} minimisant E appartient à $\mathcal{E} = \text{Span}\{e_k^i | k = 1, \dots, N; i = 1, \dots, d_k\}$.*

Démonstration. Il suffit de décomposer h sous la forme $h = h^\parallel + h^\perp$ où $h^\parallel \in \mathcal{E}$ et $h^\perp \in \mathcal{E}^\perp$. On conclut en remarquant que $E(h) = E(h^\parallel) + \|h^\perp\|^2$. \square

Ainsi, en utilisant les notations de 4.2.1, $\hat{h} = \sum_{k=1}^N e_k \cdot \alpha_k$ où les α_k sont des vecteurs de \mathbb{R}^{d_k} . Appellons S_{ij} les matrices $\langle e_i, e_j \rangle$ de taille $d_i \times d_j$ et Ξ_k matrices symétriques de taille $d_k \times d_k$ associées aux formes quadratiques ξ_k . On peut ainsi reformuler l'énergie :

$$E(h) = \sum_{ij} \alpha_i^T S_{ij} \alpha_j + \sum_k \left(\sum_i S_{ki} \alpha_i - c_k \right)^T \Xi_k \left(\sum_j S_{kj} \alpha_j - c_k \right)$$

On peut alors résoudre le problème de manière matricielle en adoptant les notations suivantes :

$$\alpha = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{pmatrix} \quad c = \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix} \quad S = \begin{pmatrix} S_{11} & \dots & S_{1N} \\ \vdots & & \vdots \\ S_{N1} & \dots & S_{NN} \end{pmatrix} \quad \Xi = \begin{pmatrix} \Xi_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Xi_N \end{pmatrix}$$

L'énergie s'écrit ainsi :

$$E(\alpha) = \alpha^T S \alpha + \sum (S \alpha - c)^T \Xi (S \alpha - c)$$

La minimisation de l'énergie peut donc s'effectuer à l'aide d'un algorithme de moindres carrés :

$$\hat{\alpha} = (S + S^T \Xi S)^{-1} S^T c$$

4.2.3 Interpolation par spline au premier ordre

Dans le contexte de l'interpolation par splines, on considère une déformation $z = id + h$ où h est un champ de vecteurs sur un ouvert $\Omega \subset \mathbb{R}^d$. On suppose que l'on a C_1 contraintes quadratiques sur z en des points x_i de Ω et C_2 contraintes quadratiques sur la différentielle $Dz = id + Dh$ en des points y_j de Ω et dans des directions u_j i.e.

$$\phi_i(z(x_i) - c_i) \approx 0, 1 \leq i \leq C_1 \quad (4.5)$$

$$\psi_j(d_{y_j} z \cdot u_j - t_j) \approx 0, 1 \leq j \leq C_2 \quad (4.6)$$

où les ϕ_i et les ψ_j sont des formes quadratiques dont les matrices de taille $d \times d$ sont respectivement Φ_i and Ψ_j .

Remarque 4.2. *Il est aussi possible de considérer des contraintes sur toute la différentielle et pas seulement dans des directions u_j . Mais cela ne nous servira pas dans nos applications.*

Afin de satisfaire ces contraintes, nous chercherons donc à minimiser une énergie du type :

$$E(h) = \|h\|^2 + \sum_{i=1}^{C_1} \phi_i(z(x_i) - c_i) + \sum_{j=1}^{C_2} \psi_j(d_{y_j} z \cdot u_j - t_j) \quad (4.7)$$

Pour relier ce problème au problème générique (4.4), nous devons exhiber un espace de Hilbert \mathcal{H} (muni d'une norme $\|\cdot\|$ et d'un produit scalaire $\langle \cdot, \cdot \rangle$) et des éléments f_{x_i}, g_{y_j, u_j} dans \mathcal{H}^d vérifiant les conditions suivantes :

$$(R) \quad \forall h \in \mathcal{H}, \forall x_i, \langle f_{x_i}, h \rangle_{\mathcal{H}} = h(x_i),$$

$$(R') \quad \forall h \in \mathcal{H}, \forall (y_j, u_j), \langle g_{y_j, u_j}, h \rangle_{\mathcal{H}} = d_{y_j} h \cdot u_j$$

Ces conditions font penser à la définition d'un espace de Hilbert à noyau reproduisant (cf. app. B), si bien qu'en choisissant un noyau arbitraire G vérifiant les conditions de la Prop. B.7 ou de la Prop. B.8, on peut construire l'espace de Hilbert associé $\mathcal{H} = \mathcal{H}^1(G)$ et les conditions (R) et (R') sont vérifiées en prenant pour tout i, j :

$$f_{x_i} = G(\cdot, x_i) \quad (4.8)$$

$$g_{y_j, u_j} = \partial_1 G(\cdot, x_j) \cdot u_j \quad (4.9)$$

Et voici la forme du champ h que nous recherchons :

$$h = \sum_i G(\cdot, x_i) \cdot \alpha_i + \sum_j (\partial_1 G(\cdot, y_j) \cdot u_j) \cdot \beta_j \quad (4.10)$$

où les α_i et les β_j sont des vecteurs de \mathbb{R}^d .

Maintenant il faut expliciter les matrices S_{ij} pour être capable de résoudre le problème de moindres carrés exposé dans 4.2.2. Ces matrices seront de trois sortes :

$$S_{1ij} = \langle f_{x_i}, f_{x_j} \rangle = G(x_i, x_j) = G(x_j, x_i)^T \quad (4.11)$$

$$S_{2ij} = \langle f_{x_i}, g_{y_j, u_j} \rangle = \partial_1 G(y_j, x_i) \cdot u_j = \partial_2 G(x_i, y_j) \cdot u_j \quad (4.12)$$

$$S_{3ij} = \langle g_{y_i, u_i}, g_{y_j, u_j} \rangle = \partial_1 \partial_2 G(y_i, y_j)(u_i, u_j) \quad (4.13)$$

Si on définit les matrices suivantes :

$$\alpha = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_{C^1} \end{pmatrix}, \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_{C^2} \end{pmatrix}, \Phi = \begin{pmatrix} \Phi_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Phi_{C^1} \end{pmatrix}, \Psi = \begin{pmatrix} \Psi_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Psi_{C^2} \end{pmatrix}$$

et si l'on appelle S_1 (resp. S_2, S_3) la matrice composée des blocs S_{1ij} (resp. S_{2ij}, S_{3ij}) alors on peut à nouveau reformuler l'énergie sous une forme matricielle plus simple :

$$\begin{aligned} E(\alpha, \beta) &= \alpha^T S_1 \alpha + \alpha^T S_2 \beta + \beta^T S_2^T \alpha + \beta^T S_3 \beta \\ &\quad + (S_1 \alpha + S_2 \beta - c)^T \Phi (S_1 \alpha + S_2 \beta - c) \\ &\quad + (S_2^T \alpha + S_3 \beta - t)^T \Psi (S_2^T \alpha + S_3 \beta - t) \end{aligned} \quad (4.14)$$

4.2.4 Algorithme de moindres carrés

Avec les notations suivantes :

$$A = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad S = \begin{pmatrix} S_1 & S_2 \\ S_2^T & S_3 \end{pmatrix} \quad \Xi = \begin{pmatrix} \Phi & 0 \\ 0 & \Psi \end{pmatrix} \quad Y = \begin{pmatrix} c - x \\ t - u \end{pmatrix}$$

on peut réécrire l'énergie comme suit :

$$E(A) = A^T S A + (S A - Y)^T \Xi (S A - Y) \quad (4.15)$$

Ainsi la minimisation est immédiate :

$$\hat{A} = (S + S^T \Xi S)^{-1} S^T \Xi Y$$

4.2.5 Prise en compte d'une transformation affine

Il est aussi possible d'inclure une transformation affine dans la déformation z . Dans ce cas, on écrit $z = id + \epsilon + h$ où ϵ est une application affine. On ajoute alors les coefficients de la transformation affine au vecteur de déformation A . En notant $\Gamma = (\gamma_{ij})$ la composante linéaire de la transformation affine et $\Delta = (\delta_i)$ la composante de translation, on peut définir un nouveau vecteur de déformation \tilde{A} :

$$\tilde{A}^T = (A^T \quad \gamma_{11} \quad \dots \quad \gamma_{1d} \quad \dots \quad \gamma_{d1} \quad \dots \quad \gamma_{dd} \quad \Delta^T)$$

On note X_i (resp. U_j) la matrice diagonale par blocs x_i^T (resp. u_j^T). Ensuite on définit deux matrices X et U :

$$X = \begin{pmatrix} X_1 & I_d \\ \vdots & \vdots \\ X_{C^1} & I_d \end{pmatrix} \quad U = \begin{pmatrix} U_1 & 0_d \\ \vdots & \vdots \\ U_{C^2} & 0_d \end{pmatrix}$$

On définit aussi la matrice suivante :

$$S_{X,U} = \begin{pmatrix} S_1 & S_2 & X \\ S_2^T & S_3 & U \end{pmatrix}$$

si bien que le vecteur $S_{X,U}\tilde{A}$ contient les points déformés et les vecteurs tangents déformés.

On veut aussi ajouter un poids $\Gamma^T\Gamma + \Delta^T\Delta$ sur la transformation affine. Puis on définit la matrice

$$S_I = \begin{pmatrix} S & 0 \\ 0 & I_{d(d+1)} \end{pmatrix}$$

et l'énergie prend alors la forme suivante :

$$E(\tilde{A}) = \tilde{A}^T S_I \tilde{A} + (S_{X,U}\tilde{A} - Y)^T \Xi (S_{X,U}\tilde{A} - Y)$$

La minimisation de l'énergie donne le résultat suivant :

$$\tilde{A} = (S_I + S_{X,U}^T \Xi S_{X,U})^{-1} S_{X,U}^T Y$$

4.3 Perspectives

Un des handicaps de la méthode exposée est qu'elle ne permet d'interpoler la déformation qu'à une seule résolution fixée par la taille du noyau utilisé. C'est-à-dire que, comme on l'a fait dans 3.1.3 pour la mise en correspondance d'images, on souhaite être capable d'analyser la déformation tout d'abord de manière grossière puis de plus en plus finement.

La solution est alors d'approximer la déformation en n'utilisant non plus un seul espace de Hilbert à noyau reproduisant mais la somme de plusieurs de ceux-ci. Si l'on considère en effet plusieurs noyaux K^1, \dots, K^N de taille décroissante, on peut obtenir une interpolation de la déformation v dans $\sum_{k=1}^N \mathcal{H}_{K^k}$ en procédant de la manière suivante :

- on interpole d'abord une déformation v_1 dans \mathcal{H}_{K^1} avec des contraintes $v_1(x_n) \simeq y_n$,
- on interpole ensuite une déformation v_2 dans \mathcal{H}_{K^2} avec des contraintes $v_2(x_n) \simeq y_n - v_1(x_n)$,
- ...
- on interpole une déformation v_k dans \mathcal{H}_{K^k} avec des contraintes $v_k(x_n) \simeq y_n - \sum_{l=1}^{k-1} v_l(x_n)$,
- ...
- on interpole enfin une déformation v_K dans \mathcal{H}_{K^K} avec des contraintes $v_K(x) \simeq y_n - \sum_{l=1}^{K-1} v_l(x_n)$.

La déformation totale $v = \sum_{k=1}^K v_k$ appartient bien à $\sum_{k=1}^K \mathcal{H}_{K^k}$. Cette méthode permet donc bien l'analyse de la déformation à plusieurs degrés de précision à la fois.

4.4 Conclusion

On a donc introduit dans ce chapitre les bases de l'interpolation par splines en les étendant à des caractéristiques de premier ordre. On verra comment faire usage de cette théorie dans les chapitres qui suivent.

Chapitre 5

Appariement et déformation

Dans ce chapitre, on s'intéressera à la fois à l'appariement de primitives et à l'estimation de la déformation entre deux ensembles de primitives. En fait, ces deux problèmes seront traités en même temps, cette méthode s'avérant plus fiable et plus robuste que leur traitement successif.

On adapte l'algorithme de mise en correspondance de points décrit dans [46] à la mise en correspondance d'objets plus généraux. On se donne deux ensembles d'objets géométriques S et T . On donnera dans un premier temps une idée de la façon dont on peut les mettre en correspondance grâce à une définition hiérarchique de ces objets et dans un cadre très général, ce qui aboutira à la définition d'une énergie de mise en correspondance (cf. 5.1). Puis on verra comment on peut minimiser l'énergie de mise en correspondance de manière à trouver l'appariement et la déformation optimaux (cf. 5.2).

On spécialisera ensuite notre algorithme à un cas concret de mise en correspondance d'objets définis par des points et des vecteurs (tangentes ou normales) (cf. 5.4) et on donnera quelques résultats en ce qui concerne la mise en correspondance de segments, de contours ou de facettes (cf. 5.5).

5.1 Formalisation du problème et définition d'une énergie de mise en correspondance

5.1.1 Modélisation hiérarchique des ensembles

Un ensemble est défini de manière hiérarchique comme un assemblage de composantes elles-mêmes divisés en "sous-composantes", encore une fois composées de "sous-sous-composantes", etc... Le but est alors de mettre en correspondance chaque niveau hiérarchique des ensembles. La figure 5.1 représente de manière schématique la structure hiérarchique de deux ensembles ainsi que leur mise en correspondance.

Soient deux ensembles S et T deux ensembles hiérarchiques à mettre en correspondance. On note S^h (resp. T^h) le niveau h de la hiérarchie de S (resp. T) et S^* (resp. T^*) le dernier niveau de la hiérarchie de S (resp. T). On suppose que S et T possèdent le même nombre de niveaux hiérarchiques. Enfin on note \mathcal{E} l'ensemble auquel appartiennent les composantes de S^* et T^* . Si l'on considère l'ensemble S comme un arbre, S^h est l'ensemble des noeuds à la profondeur h

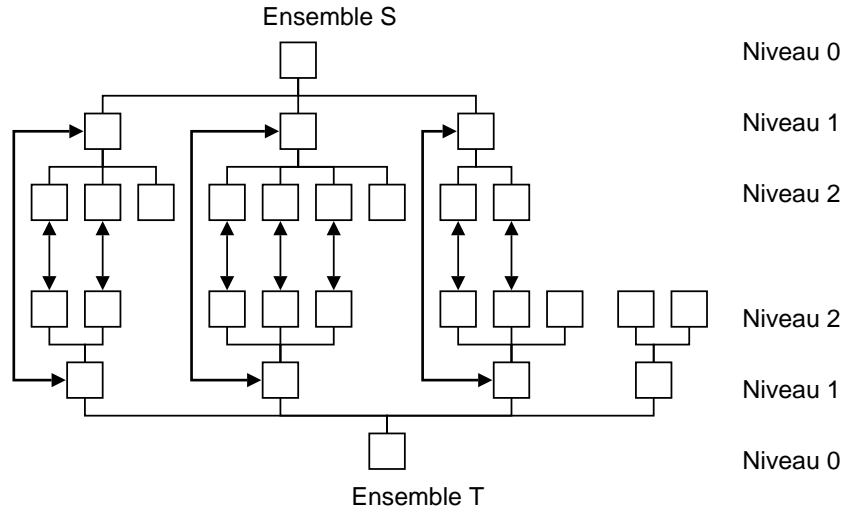


FIG. 5.1 – Mise en correspondance hiérarchique de deux ensembles

et S^* est l'ensemble des feuilles de l'arbre.

Par exemple, un ensemble peut être composé de plusieurs objets 3D, chaque objet étant lui-même composé de facettes 3D et chaque facette étant défini par un certain nombre de points 3D définissant son contour. Le niveau 0 est alors l'ensemble lui-même, le niveau 1 l'ensemble des objets 3D, le niveau 2 l'ensemble des facettes et le niveau 3 l'ensemble des points 3D.

On définit une application qui permet de passer d'un niveau hiérarchique au niveau hiérarchique supérieur.

Définition 5.1. On note r l'application qui à une composante σ d'un niveau hiérarchique S^h d'un ensemble S fait correspondre sa "composante-mère" $s = r(\sigma) \in S^{h-1}$. On note pour $h \in \mathbb{N}$ r^h la composée de h fois l'application r avec la convention $r^0 = Id$.

5.1.2 Modélisation des appariements

Plaçons-nous au niveau h de la hiérarchie. Prenons une composante s de niveau h dans l'ensemble S et une composante t de niveau h dans l'ensemble T . A moins que le niveau h soit le dernier niveau de la hiérarchie, s et t sont respectivement divisées en K_s sous-composantes σ et N_t sous-composantes τ . Les correspondances entre les sous-composantes σ et τ sont modélisées par une matrice binaire M^{st} de taille $(K_s + 1) \times (N_t + 1)$ (la dernière colonne et la dernière ligne permettent de prendre en compte les outliers) de la forme suivante :

	correspondances					outlier
correspondances	0	1	0	0	0	0
	0	0	0	0	0	1
	1	0	0	0	0	0
outlier	0	0	1	1	1	

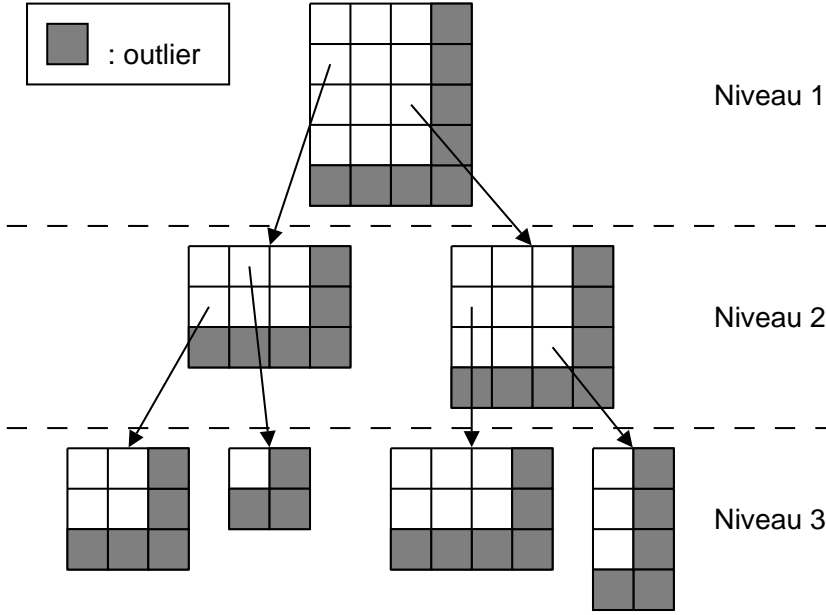


FIG. 5.2 – Hiérarchie des matrices d'appariement

La hiérarchie des ensembles S et T définit donc une hiérarchie de matrices d'appariement comme l'illustre la figure 5.2.

Néanmoins, les coefficients des matrices d'appariement sont autorisés à prendre des valeurs continues entre 0 et 1 afin de relâcher les contraintes du problème, même si en fin de compte, on recherche une correspondance biunivoque. Les sommes de chaque ligne et de chaque colonne de la matrice (exceptées la dernière ligne et la dernière colonne) doivent être égales à 1¹ :

$$\forall \tau \in t, \sum_{\sigma \in s, \sigma \neq \emptyset} M_{\sigma\tau}^{st} = 1, \quad \forall \sigma \in s, \sum_{\tau \in t, \tau \neq \emptyset} M_{\sigma\tau}^{st} = 1 \quad (5.1)$$

Les coefficients de la matrice M^{st} peuvent aussi être interprétés de manière probabiliste, chaque coefficient $M_{\sigma\tau}^{st}$ représentant la probabilité que les sous-composantes $\sigma \in s$ et $\tau \in t$ soient appariées sachant que les composantes s et t sont appariées. On notera \leftrightarrow la relation d'équivalence qui exprime le fait que deux composantes de même niveau hiérarchique sont appariées. De plus $\sigma \leftrightarrow \tau$ implique que $r(\sigma) \leftrightarrow r(\tau)$. Ainsi, la loi de Bayes nous permet d'écrire pour $\sigma \in S^h$ et $\tau \in T^h$:

$$P(\sigma \leftrightarrow \tau) = P(r(\sigma) \leftrightarrow r(\tau), \sigma \leftrightarrow \tau) = P(r(\sigma) \leftrightarrow r(\tau)) M_{\sigma\tau}^{r(\sigma)r(\tau)} \quad (5.2)$$

Proposition 5.2. *Les probabilités d'appariement des composantes peuvent s'exprimer uniquement en fonction des matrices d'appariement. Pour être plus précis, on a la relation, pour $\sigma \in S^h$ et $\tau \in T^h$:*

$$P(\sigma \leftrightarrow \tau) = \prod_{k=0}^{h-1} M_{r^k(\sigma)r^k(\tau)}^{r^{k+1}(\sigma)r^{k+1}(\tau)}$$

¹Par la suite, on confondra, les indices des matrices et les composantes qu'ils représentent et le symbole \emptyset représentera les outliers.

Démonstration. Il suffit d'écrire la relation (5.2) pour les niveaux k allant de h à 1. \square

Par la suite, on aura besoin de l'entropie globale du système afin de caractériser le degré de précision de l'appariement. On définit d'abord l'entropie d'une matrice d'appariement.

Définition 5.3. *Pour une matrice d'appariement M^{st} vérifiant la relation (5.1), on définit l'entropie $H(M^{st})$ de cette matrice par la relation :*

$$H(M^{st}) = - \sum_{\substack{\sigma \in s \cup \emptyset \\ \tau \in t \cup \emptyset}} M_{\sigma\tau}^{st} \log(M_{\sigma\tau}^{st})$$

On peut alors énoncer la proposition suivante.

Proposition 5.4. *L'entropie globale du système peut s'écrire en fonction des seules matrices d'appariement :*

$$H(S, T, M) = \sum_{h=1}^K \sum_{s \in S^h, t \in T^h} \prod_{h'=0}^h M_{r^{h'+1}(s)r^{h'+1}(t)}^{r^{h'+1}(s)r^{h'+1}(t)} H(M^{st}) \quad (5.3)$$

où K est le nombre total de niveaux hiérarchiques.

Démonstration. Le système est totalement caractérisé par les correspondances du niveau hiérarchique le plus bas puisque $s \leftrightarrow t \Rightarrow r(s) \leftrightarrow r(t)$. L'entropie globale s'exprime donc de la manière suivante :

$$\begin{aligned} H(S, T, M) &= H(P(\sigma \leftrightarrow \tau, \sigma \in S^h \cup \emptyset, \tau \in T^h \cup \emptyset, 0 \leq h \leq K)) \\ &= H(P(\sigma \leftrightarrow \tau, \sigma \in S^* \cup \emptyset, \tau \in T^* \cup \emptyset)) \end{aligned}$$

Or la relation (5.2) implique :

$$\begin{aligned} H(P(\sigma \leftrightarrow \tau, \sigma \in S^* \cup \emptyset, \tau \in T^* \cup \emptyset)) &= H(P(s \leftrightarrow t, s \in S^{K-1} \cup \emptyset, t \in T^{K-1} \cup \emptyset)) \\ &\quad + \sum_{\substack{s \in S^{K-1} \cup \emptyset \\ t \in T^{K-1} \cup \emptyset}} P(s \leftrightarrow t) H(M^{st}) \end{aligned}$$

On répète le même raisonnement pour calculer $H(P(s \leftrightarrow t, s \in S^{K-1} \cup \emptyset, t \in T^{K-1} \cup \emptyset))$ et ainsi de suite. En répétant ce calcul $K-1$ fois, on obtient finalement l'expression (5.3). \square

5.1.3 Modélisation de la déformation

Pour l'instant, on définit la déformation f comme une action sur \mathcal{E} . L'ensemble \mathcal{E} étant pour l'instant défini de manière purement formelle, il est pour l'instant difficile de donner une définition plus précise de l'ensemble des déformations. Mais on donnera un exemple d'un type de déformation dans 5.4.3.

5.1.4 Définition d'une énergie de mise en correspondance

On donne une expression de l'énergie avec un terme de mise en correspondance, un terme de régularisation de la déformation et un terme de précision de l'appariement.

$$\begin{aligned}
E^\theta(S, T, f, M) = & \sum_{\sigma \in S^*, \tau \in T^*} P(\sigma \leftrightarrow \tau) D_{\sigma\tau}(f(\sigma), \tau) \\
& + \theta H(P(\sigma \leftrightarrow \tau), \sigma \in S^* \cup \emptyset, \tau \in T^* \cup \emptyset) \\
& - \kappa \sum_{\sigma \in S^*, \tau \in T^*} P(\sigma \leftrightarrow \tau) + N(f)
\end{aligned} \tag{5.4}$$

où les $D_{\sigma\tau}$ sont des distances sur \mathcal{E} et N est une norme régularisante. Le paramètre de température θ contrôle le degré d'indécision de la mise en correspondance. Le terme en κ empêche le minimum global d'être atteint pour la configuration "tous outliers". En effet, sans ce terme, on aurait $E^\theta(S, T, f, M) \geq 0$ et on vérifie sans problème que la configuration "tous outliers" correspondrait dans ce cas à une énergie nulle.

5.2 Minimisation de l'énergie

La minimisation de l'énergie consiste en l'itération de deux étapes – une étape de "soft assign"(cf. [29]) mettant à jour les correspondances et une étape de déformation – incluse dans un recuit déterministe. Il s'agit en fait d'un algorithme de type "EM" (Expectation Minimization) [24, 44] où les distributions de probabilités à évaluer sont les correspondances entre les éléments des ensembles à appairer et où le paramètre inconnu est la déformation entre les deux ensembles.

5.2.1 Mise à jour des correspondances

On minimise l'énergie avec la déformation f fixée. Il faut donc trouver les correspondances minimisant l'énergie. Pour cela, on évalue d'abord les correspondances en minimisant l'énergie sans tenir compte des contraintes de double stochasticité sur les matrices M^{ST} (cf. (5.1)) puis on impose ces contraintes en utilisant un algorithme de Sinkhorn (cf. Alg. 2) qui consiste à normaliser itérativement les colonnes puis les lignes des matrices M^{ST} jusqu'à convergence (la convergence de cet algorithme est prouvée dans [52]).

En prenant les notations suivantes pour le niveau hiérarchique le plus bas (i.e. $\sigma \in S^*, \tau \in T^*$) :

$$d_{\sigma\tau} = D_{\sigma\tau}(f(\sigma), \tau) \tag{5.5}$$

$$a_{\sigma\tau} = 1 \tag{5.6}$$

$$h_{\sigma\tau} = 0 \tag{5.7}$$

et en les étendant aux niveaux supérieurs de la manière suivante :

$$d_{ST} = \sum_{s \in S, t \in T} M_{st}^{ST} d_{st} \quad (5.8)$$

$$a_{ST} = \sum_{s \in S, t \in T} M_{st}^{ST} a_{st} \quad (5.9)$$

$$h_{ST} = H(M^{ST}) + \sum_{s \in S, t \in T} M_{st}^{ST} h_{st} \quad (5.10)$$

on peut montrer que les matrices d'appariement d'un certain niveau hiérarchique h – les matrices d'appariement des autres niveaux étant fixées – minimisent l'énergie si :

$$\hat{M}_{st}^{ST} = \exp \left(\frac{\kappa a_{st} - d_{st}}{T} + h_{st} \right)$$

Il ne reste alors plus qu'à normaliser les matrices en utilisant l'algorithme de Sinkhorn.

La décroissance de l'énergie à la suite de cette procédure se prouve en utilisant la positivité de la distance de Kullback-Leibler et la double stochasticté des matrices M^{ST} avant et après la mise à jour.

Algorithm 2 Algorithme de Sinkhorn pour une matrice M de taille $(K+1, L+1)$ (outliers : dernière ligne et dernière colonne)

repeat

Normaliser à 1 les K premières lignes de M

Normaliser à 1 les L premières colonnes de M

until $\left\{ \begin{array}{l} \forall k \in \{1, \dots, K\}, \quad \left| \sum_{l=1}^{L+1} M_{kl} - 1 \right| < \epsilon \\ \forall l \in \{1, \dots, L\}, \quad \left| \sum_{k=1}^{K+1} M_{kl} - 1 \right| < \epsilon \end{array} \right.$

On peut donc mettre à jour les matrices d'appariement “en cascade” du niveau hiérarchique le plus bas au niveau le plus haut (cf. Alg. 3).

5.2.2 Mise à jour de la déformation

La déformation f est pour l'instant définie de manière trop abstraite pour que l'on puisse minimiser formellement l'énergie en f . Néanmoins, dans la section suivante f sera définie de manière plus concrète et nous verrons donc un exemple de minimisation dans la section 5.4.6.

5.2.3 Algorithme générique

Les étapes de mise à jour des correspondances et de la déformation sont incluses dans un processus de recuit déterministe contrôlé par la température θ (cf. Alg. 4).

Le paramètre κ a une interprétation physique. A faible température, il agit en effet comme un seuil : lorsque θ tend vers 0, les coefficients $\hat{M}_{\sigma\tau}^{r(\sigma)r(\tau)}$ pour

Algorithm 3 Mise à jour en cascade des correspondances

for $\sigma \in S^K, \tau \in T^K$ **do**

$$d_{\sigma\tau} \leftarrow D_{\sigma\tau}(f(\sigma), \tau)$$

$$a_{\sigma\tau} \leftarrow 1$$

$$h_{\sigma\tau} \leftarrow 0$$

$$\hat{M}_{\sigma\tau}^{r(\sigma)r(\tau)} \leftarrow \exp\left(\frac{\kappa a_{\sigma\tau} - d_{\sigma\tau}}{\theta} + h_{\sigma\tau}\right)$$

end for
for $s \in S^{K-1}, \tau \in T^{K-1}$ **do**

 Normaliser les matrices M^{st}
end for
for ($h = K - 1; h > 0; h - -$) **do**
for $s \in S^h, t \in T^h$ **do**

$$d_{st} \leftarrow \sum_{\sigma \in s, \tau \in t} M_{\sigma\tau}^{st} d_{\sigma\tau}$$

$$a_{st} \leftarrow \sum_{\sigma \in s, \tau \in t} M_{\sigma\tau}^{st} d_{\sigma\tau}$$

$$h_{st} \leftarrow H(M^{st}) + \sum_{\sigma \in s, \tau \in t} M_{\sigma\tau}^{st} h_{\sigma\tau}$$

$$\hat{M}_{st}^{r(s)r(t)} \leftarrow \exp\left(\frac{\kappa a_{st} - d_{st}}{\theta} + h_{st}\right)$$

end for
for $S \in S^{h-1}, T \in T^{h-1}$ **do**

 Normaliser les matrices M^{ST}
end for
end for

Algorithm 4 Mise en correspondance hiérarchique

repeat
for ($i := 0; i < N_{iter}; i + +$) **do**

Mise à jour des correspondances par soft assign hiérarchique

Mise à jour de la déformation par moindres carrés

end for

 Diminuer la température θ
until $\theta < \theta_{min}$

$\sigma \in S^*, \tau \in T^*$ se comportent de la manière suivante :

$$\begin{aligned} D_{\sigma\tau}(f(\sigma), \tau) < \kappa &\Rightarrow \hat{M}_{\sigma\tau}^{r(\sigma)r(\tau)} \gg 1 \\ D_{\sigma\tau}(f(\sigma), \tau) > \kappa &\Rightarrow \hat{M}_{\sigma\tau}^{r(\sigma)r(\tau)} \ll 1 \end{aligned}$$

Les paramètres κ peuvent donc être réglés en fonction de la précision souhaitée *a posteriori* sur la mise en correspondance finale.

Au contraire, la température initiale peut être fixée à partir de la connaissance *a priori* sur la proximité des deux ensembles à mettre en correspondance. Si l'on a pas de connaissance *a priori* et que l'on souhaite que le recuit déterministe puisse explorer toutes les hypothèses de mise en correspondance alors il faut poser :

$$\theta^0 = \max_{\sigma \in S^*, \tau \in T^*} \{D(f^0(\sigma), \tau)\}$$

où f^0 est l'initialisation de la déformation ($f^0 = Id$ en général). En effet, comme $\hat{M}_{\sigma\tau}^{r(\sigma)r(\tau)} = \exp\left(\frac{\kappa - D(f(\sigma), \tau)}{\theta}\right)$, la température θ^0 est assez élevée pour ne pas privilégier certains appariements *a priori* au début de l'algorithme.

5.3 Etude d'un cas simple : appariement de points en multi-vues

On va étudier dans un premier temps un cas simple pour illustrer le processus de "soft assign". On se place dans les conditions suivantes :

- on considère un seul niveau hiérarchique,
- on ne prend pas en compte de déformation.

Contexte

On va profiter de la diversité de contraintes introduite par les formes quadratiques dans le chapitre 4 pour l'appariement de points en multivues. On suppose que l'on dispose de plusieurs images I_n d'une même scène dont on connaît les conditions de prise de vue. Ceci est bien le cas pour les images de nos expériences qui nous ont été fournies par l'IGN avec leurs conditions de prises de vue.

On a extrait de chaque image I_n un ensemble de points $\{Q_b^n\}$ et l'on veut trouver les correspondances entre les différents ensembles de points Q^n ainsi que les points $\{P_a\}$ de la scène 3D correspondant aux points 2D. On verra que l'on peut effectuer ces deux tâches en même temps. D'ailleurs l'estimation des points 3D servira à la mise en correspondance des points 2D et vice-versa (comme dans le chapitre 5, l'appariement et l'estimation de la déformation étaient effectués en même temps et se contraignaient l'une l'autre).

Dans les problèmes de multi-vues, on veut généralement éviter de prendre une des images comme image de référence afin d'éviter toute sorte de biais. On adopte donc la notion de "true multi-image matching" introduite dans [22]. La solution est alors de prendre l'espace objet (3D) comme référence. C'est ce que nous allons faire ici. On estimera non pas des appariements entre les différents ensembles de points Q^n (ce qui nécessiterait $\frac{N(N-1)}{2}$ mises en correspondance)

mais des appariements entre l'ensemble de points 3D P et les différents ensembles de points 2D Q^n : on a donc en fait N problèmes de mise en correspondance liés entre eux.

Le cadre théorique est donc légèrement différent de celui de la section 5.1. On a un seul niveau hiérarchique à savoir les points 2D et 3D et N matrices de correspondance. De plus, il n'y a pas de déformation inconnue à retrouver mais les points 3D P_a seront eux-mêmes des inconnues du problème.

Définitions

On appellera M^n la matrice d'appariement entre l'ensemble P et l'ensemble Q^n (cf. 5.1.2). A noter que les outliers permettent de prendre en compte les problèmes liés aux occlusions. On appellera F^n le sommet de prise de vue de l'image I_n . On définit alors l'énergie de mise en correspondance suivante :

$$E^\theta(M, P) = \sum_{n,a,b} M_{ab}^n d(P_a, F^n Q_b^n)^2 + \theta \sum_{n,a,b} M_{ab}^n \log M_{ab}^n$$

où $d(P_a, F^n Q_b^n)$ est la distance du point P_a à la droite $F^n Q_b^n$ (cf. fig. 5.3). Les conditions de prise de vue étant connues dans chaque image, la quantité $d(P_a, F^n Q_b^n)$ ne dépend donc que de l'inconnue P_a .

Si on note u_a^n le vecteur directeur unitaire de la droite $F^n Q_a^n$ alors l'énergie peut s'écrire :

$$E^\theta(M, P) = \sum_{n,a,b} M_{ab}^n \left\| \overrightarrow{F^n P_a} \wedge u_b^n \right\|^2 + \theta \sum_{n,a,b} M_{ab}^n \log M_{ab}^n - \kappa \sum_{n,a,b} M_{ab}^n$$

Ainsi on constate que l'énergie est quadratique en les inconnues P_a . Si l'on note Φ_b^n la matrice de la forme quadratique² $\phi_b^n : x \rightarrow \|x \wedge u_b^n\|^2$, on peut réécrire l'énergie sous forme matricielle :

$$E^\theta(M, P) = \sum_{n,a,b} M_{ab}^n (P_a - F^n)^T \Phi_a^n (P_a - F^n) + \theta \sum_{n,a,b} M_{ab}^n \log M_{ab}^n - \kappa \sum_{n,a,b} M_{ab}^n$$

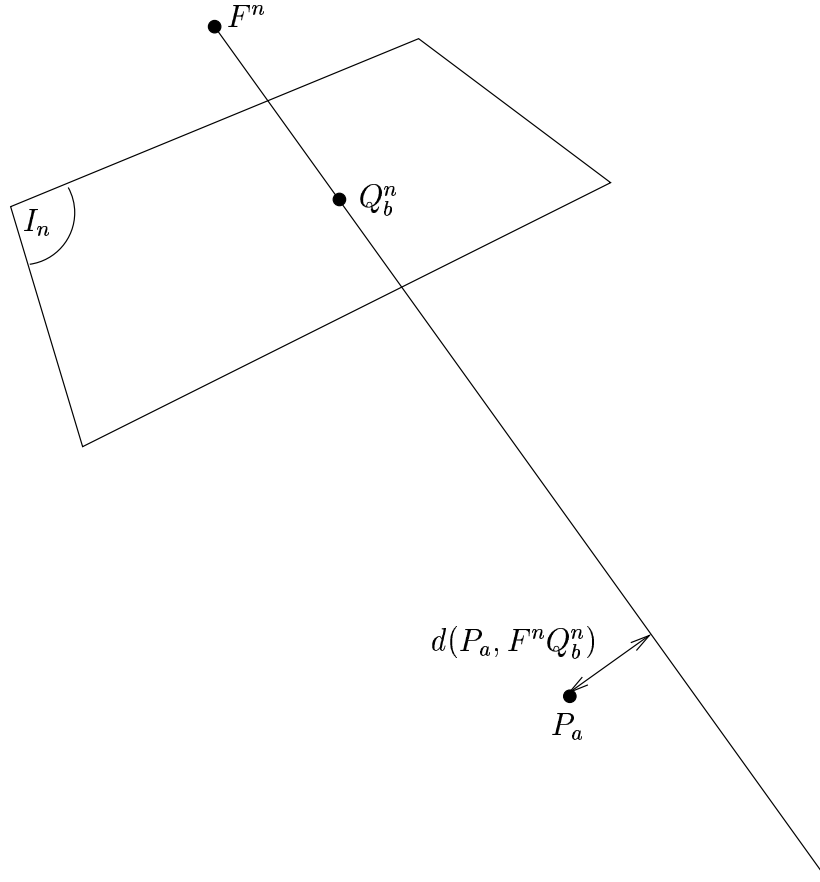
Algorithme de minimisation de l'énergie

On a à nouveau un algorithme avec l'itération de deux étapes de mise à jour – l'une pour les matrices d'appariement, l'autre pour les points 3D – imbriquée dans un processus de recuit déterministe (cf. Alg. 4). On notera que dans le cas présent, la mise à jour de la déformation est remplacée par la mise à jour des points 3D. La mise à jour des matrices d'appariement est une simple application de l'algorithme 3 au cas où on a un seul niveau hiérarchique. La mise à jour des points 3D correspond à la résolution de $\frac{\partial E}{\partial P_a} = 0$ pour tout a . Ces mises à jour sont très simples dans ce cas et on donne directement le résultat dans Alg. 5.

Résultats

Il faut noter que l'algorithme nécessite la donnée *a priori* du nombre de points 3D. Ceci n'est pas un handicap : il suffit de surévaluer le nombre de points 3D, les points surnuméraires étant comptabilisés comme outliers.

²A la différence du cadre plus général introduit dans le chapitre 4, les formes quadratiques sont ici indexées seulement sur les éléments de l'ensemble d'arrivée i.e. les points 2D Q_b^n .

FIG. 5.3 – Distance du point P_a à la droite $F^n Q_b^n$

Algorithm 5 Mise en correspondance de points en multivues

repeat

Mise à jour des appariements

$$\hat{M}_{ab}^n = \exp\left(\frac{\kappa - \phi_a^n(P_a - F^n)}{\theta}\right)$$

puis normalisation itérée sur les lignes et les colonnes,

Mise à jour des points 3D :

$$\forall a, \frac{\partial E}{\partial P_a} = 0 \Leftrightarrow \sum_{n,b} M_{ab}^n \Phi_b^n (P_a - F^n) = 0$$

$$\Leftrightarrow P_a = \left(\sum_{n,b} M_{ab}^n \Phi_b^n \right)^{-1} \sum_{n,b} M_{ab}^n \Phi_b^n F^n$$

Diminuer θ **until** $\theta < \theta_f$

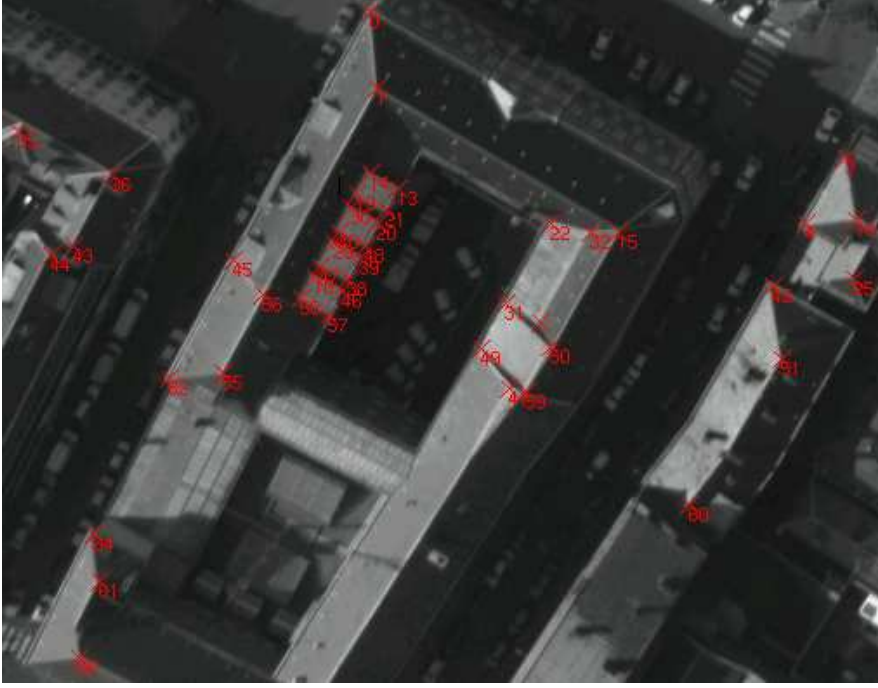


FIG. 5.4 – Appariement de points en multi-vues : image 1. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.

Dans les figures 5.4 à 5.9, les numéros des points sont les indices des points 3D avec lesquels sont mis en correspondance les points des images. Ainsi, des points de différentes images ayant le même numéro sont en correspondance avec le même point 3D et donc en correspondance entre eux.

5.4 Etude d'un cas plus complexe

Nous nous limiterons dans cette section à deux niveaux hiérarchiques. Les composantes du dernier niveau seront des points et des vecteurs. Les composantes de niveau 1 (i.e. les objets) seront donc des assemblages de points et de vecteurs (composantes de niveau 2).

5.4.1 Définition des objets

On suppose que chaque objet du premier ensemble $\{S_a\}$ est défini par K_a points $(p_a^i)_{1 \leq i \leq K_a}$ et K'_a directions $(u_a^j)_{1 \leq j \leq K'_a}$ en des points $(x_a^j)_{1 \leq j \leq K'_a}$. Les composantes des objets $\{T_b\}$ ne sont pas à proprement parler des points et des vecteurs au sens géométriques du terme mais des paramètres entrant en compte dans la définition des contraintes. Ainsi, pour tout $b \in \{1, \dots, N\}$, on construit des vecteurs $\{c_b^j, 1 \leq j \leq N_b\}$ et $\{t_b^l, 1 \leq l \leq N'_b\}$ (voir équation (5.12)) qui n'ont pas forcément d'interprétation géométrique directe. On construit aussi des formes quadratiques ϕ_{ab}^{ij} and ψ_{ab}^{kl} pour imposer le type de mise en correspondance souhaité (des exemples concrets seront donnés dans 5.5).

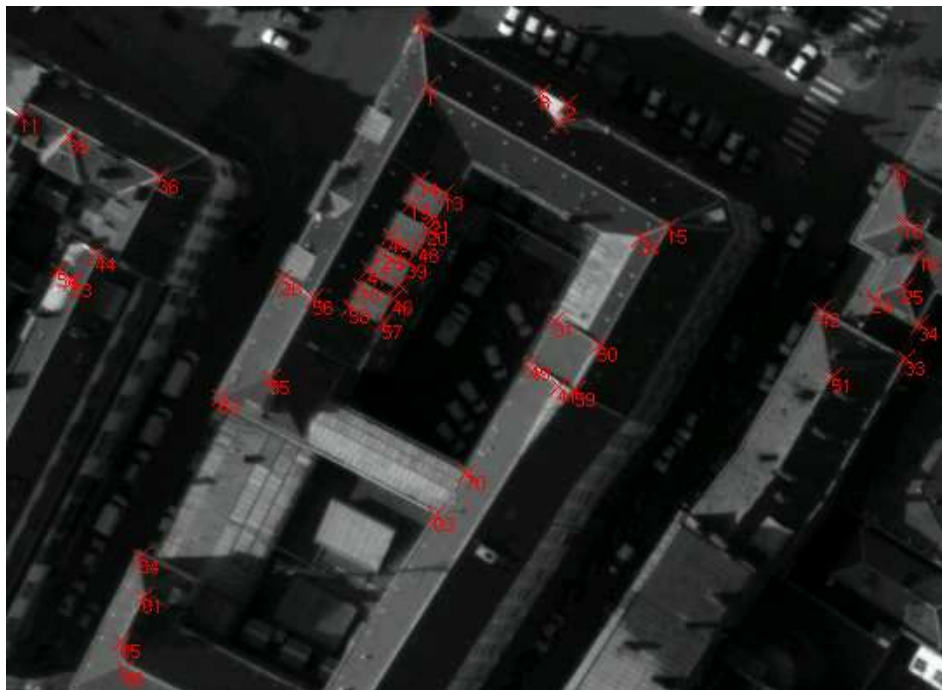


FIG. 5.5 – Appariement de points en multi-vues : image 2. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.

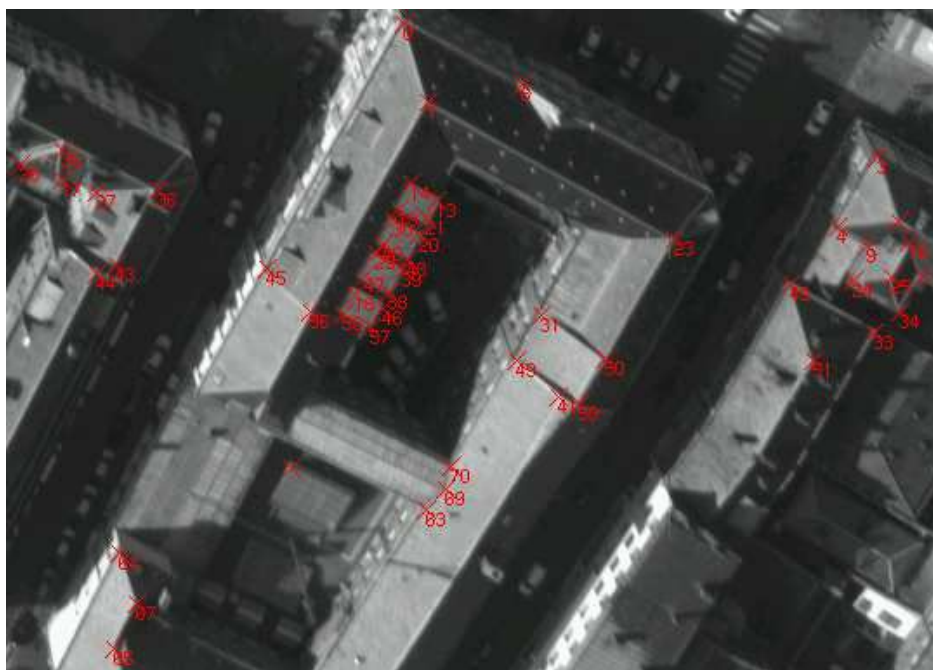


FIG. 5.6 – Appariement de points en multi-vues : image 3. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.

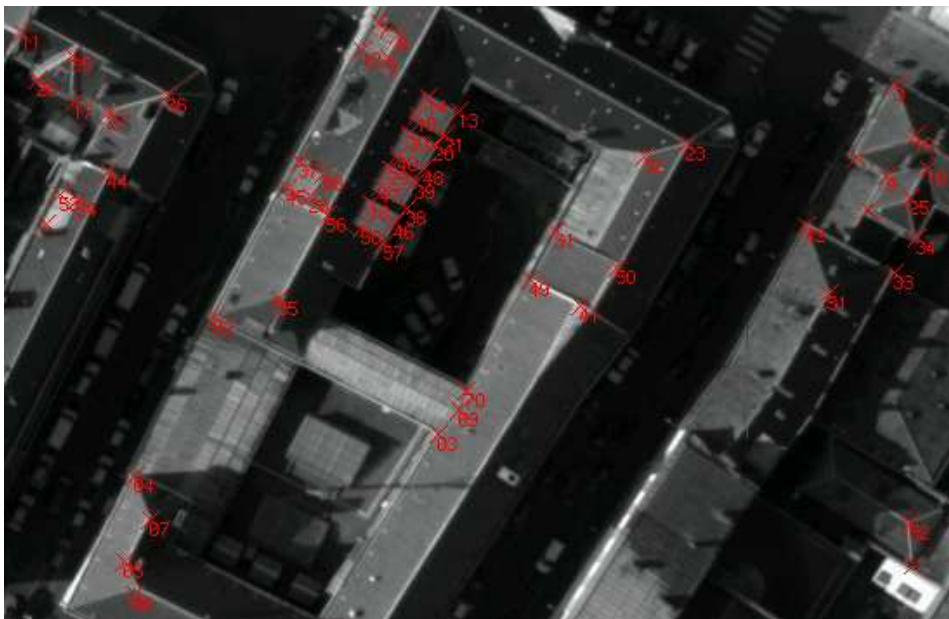


FIG. 5.7 – Appariement de points en multi-vues : image 4. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.

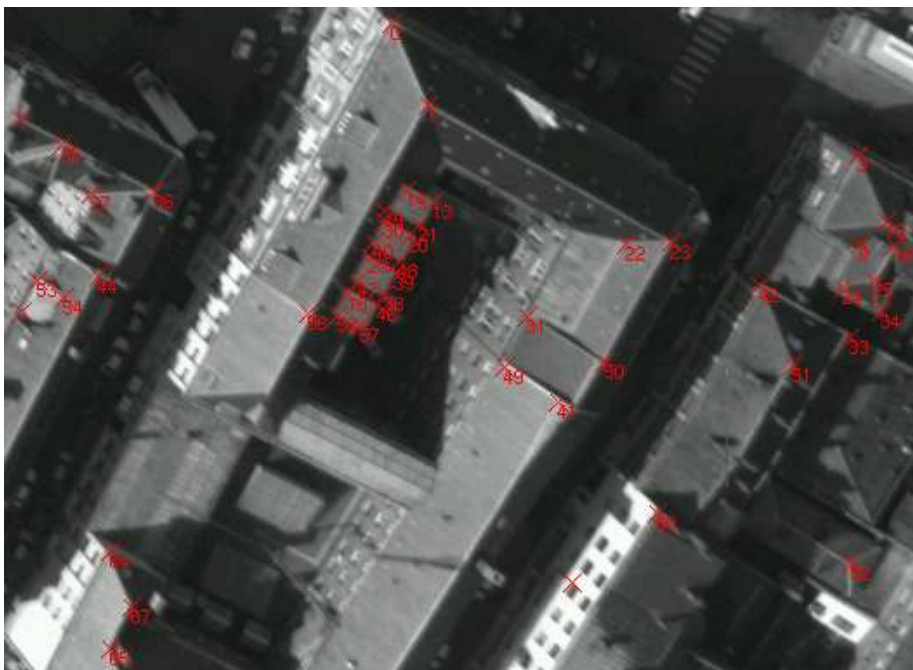


FIG. 5.8 – Appariement de points en multi-vues : image 5. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.

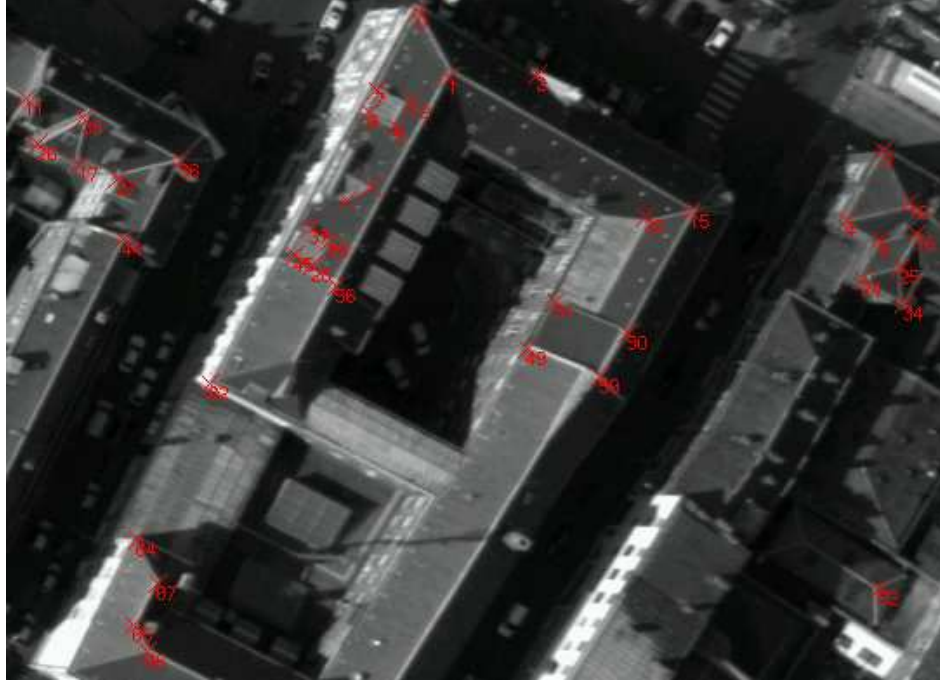


FIG. 5.9 – Appariement de points en multi-vues : image 6. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.

5.4.2 Matrices d'appariement

On a seulement deux niveaux hiérarchiques. On appellera M la matrice d'appariement de niveau 1. La situation est un peu différente du cadre général puisque dans ce cas particulier, on veut appairer des composantes de même type (i.e. des points avec des points, des vecteurs avec des vecteurs). On aura donc deux types de matrice de “sous-correspondance” : P_{ab} pour les correspondances de points et Q_{ab} pour les correspondances de vecteurs.

On peut alors reformuler l'entropie globale du système (cf. eqn. (5.3)) de la façon suivante :

$$\begin{aligned}
 H(M, P, Q) &= H(M) + \sum_{a,b} M_{ab} H(P, Q | M = (a, b)) \\
 &= H(M) + \sum_{a,b} M_{ab} H(P | M = (a, b)) + \sum_{a,b} M_{ab} H(Q | M = (a, b)) \\
 &= H(M) + \sum_{a,b} M_{ab} H(P_{ab}) + \sum_{a,b} M_{ab} H(Q_{ab})
 \end{aligned} \tag{5.11}$$

5.4.3 Espace des déformations

Les composantes des objets que l'on mettra en correspondance seront des points et des vecteurs. La déformation sera donc modélisée comme dans la section 4.2.5 par une application $f = \epsilon + h$ où h est un champ de vecteurs sur un

ouvert $\Omega \subset \mathbb{R}^d$ appartenant à un certain Hilbert \mathcal{H} à noyau reproduisant G de premier ordre (cf. app. B) et ϵ une application affine.

5.4.4 Energie de mise en correspondance

Après ces définitions, il est maintenant possible de spécialiser l'énergie de mise en correspondance (5.4) au cas qui nous intéresse.

$$\begin{aligned}
E^\theta(S, T) = & \sum_{a=1}^K \sum_{b=1}^N M_{ab} \left(\sum_{i=1}^{K_a} \sum_{j=1}^{N_b} P_{ab}^{ij} \phi_{ab}^{ij} (c_b^j - f(p_a^i)) \right) \\
& + \sum_{a=1}^K \sum_{b=1}^N M_{ab} \left(\sum_{k=1}^{K'_a} \sum_{l=1}^{N'_b} Q_{ab}^{kl} \psi_{ab}^{kl} (t_b^l - D_{x_a^k} f(u_a^k)) \right) \\
& + \theta \left(\sum_{ab} M_{ab} \log M_{ab} + \sum_{ab} M_{ab} \left(\sum_{ij} P_{ab}^{ij} \log(P_{ab}^{ij}) + \sum_{kl} Q_{ab}^{kl} \log(Q_{ab}^{kl}) \right) \right) \\
& - \kappa_1 \sum_{abij} M_{ab} P_{ab}^{ij} - \kappa_2 \sum_{abkl} M_{ab} Q_{ab}^{kl} + \|h\|_{\mathcal{H}}^2 + \|\epsilon\|^2
\end{aligned} \tag{5.12}$$

A nouveau, le champ de vecteurs h minimisant E doit être de la forme décrite dans (4.10) :

$$h = \sum_{a,i} G(p_a^i, \cdot) \cdot \alpha_a^i + \sum_{a,j} (\partial_1 G(x_a^j, \cdot) \cdot u_a^j) \cdot \beta_a^j \tag{5.13}$$

5.4.5 Etapes d'appariement

Puisque, dans ce cas particulier, n'y a plus que deux niveaux hiérarchiques de matrices d'appariement, la mise à jour en cascade des matrices de correspondance (cf. sec. 5.2.1 et Alg. 3) ne comporte plus que deux étapes :

- une mise à jour *locale* des correspondances entre points et vecteurs,
- une mise à jour *globale* des correspondances entre objets.

Etape de soft assign local

Cette étape met à jour les matrices de "sous-correspondance" Q_{ab} et P_{ab} d'après la déformation courante et la matrice de correspondance globale M . Pour tout a, b , on fixe la matrice P_{ab} (resp. Q_{ab}) égale à la matrice \hat{P}_{ab} (resp. \hat{Q}_{ab}) telle que $\frac{\partial E}{\partial P_{ab}}(\hat{P}_{ab}) = 0$ (resp. $\frac{\partial E}{\partial Q_{ab}}(\hat{Q}_{ab}) = 0$). Ce qui donne pour tout a, b, i, j, k, l :

$$\hat{P}_{ab}^{ij} = \exp \left(\frac{\kappa_1 - \phi_{ab}^{ij} (c_b^j - f(p_a^i))}{\theta_1} \right) \tag{5.14}$$

$$\hat{Q}_{ab}^{kl} = \exp \left(\frac{\kappa_2 - \psi_{ab}^{kl} (t_b^l - d_{x_a^k} f(u_a^k))}{\theta_2} \right) \tag{5.15}$$

Ensuite il faut normaliser \hat{P}_{ab} (resp. \hat{Q}_{ab}) de telle sorte que ses lignes et ses colonnes aient pour somme 1. Pour ce faire, on utilise l'algorithme Alg. 2 pour les matrices \hat{P}_{ab} et \hat{Q}_{ab} .

Etape de soft assign global

Cette étape met à jour la matrice d'appariement en tenant compte de la déformation courante et des matrices de sous-correspondance courantes. D'abord, la matrice d'appariement est choisie comme la matrice \hat{M} minimisant l'énergie (sans tenir compte des contraintes de stochasticité) i.e. telle que $\frac{\partial E}{\partial M}(\hat{M}) = 0$. Cela donne pour tout a, b :

$$\hat{M}_{ab} = \exp \left(\frac{\kappa_1 s_{ab}^1 + \kappa_2 s_{ab}^2 - d_{ab}^1 - d_{ab}^2}{\theta} + h_{ab}^1 + h_{ab}^2 \right)$$

avec

$$\begin{aligned} h_{ab}^1 &= \sum_{ij} P_{ab}^{ij} \log P_{ab}^{ij} & s_{ab}^1 &= \sum_{ij} P_{ab}^{ij} & d_{ab}^1 &= \sum_{ij} P_{ab}^{ij} \phi_{ab}^{ij}(c_b^i - f(p_a^i)) \\ h_{ab}^2 &= \sum_{kl} Q_{ab}^{kl} \log Q_{ab}^{kl} & s_{ab}^2 &= \sum_{kl} Q_{ab}^{kl} & d_{ab}^2 &= \sum_{kl} Q_{ab}^{kl} \psi_{ab}^{kl}(t_b^l - d_{x_a^i} f(u_a^k)) \end{aligned}$$

La matrice \hat{M} est ensuite aussi normalisée par l'algorithme de Sinkhorn (cf. Alg. 2).

5.4.6 Etape de déformation

On veut la déformation f connaissant les correspondances courantes, c'est-à-dire qu'on cherche les coefficients α_a^i et η_a^j de l'expression (5.13) ainsi que la matrice $\Gamma = (\gamma_{ij})$ et le vecteur Δ relatifs à la partie affine ϵ de la déformation. On montrera que ce problème revient à un problème de moindres carrés similaire à celui traité dans 4.2.4.

On doit à nouveau définir une matrice de raideur comme dans (4.11). Dans un but de simplification, nous noterons $G'(x, y, z) = \partial_2 G(x, y) \cdot z$ et $G''(x, y, z, t) = \partial_1 \partial_2 G(x, y)(z, t)$. Pour tout $(a, a') \in \{1, \dots, K\}^2$, définissons les matrices :

$$\begin{aligned} - S_{aa'}^1 &= \left(G(p_a^i, p_{a'}^j) \right)_{1 \leq i, j \leq C_1}, \\ - S_{aa'}^2 &= \left(G'(p_a^i, x_{a'}^j, u_{a'}^j) \right) \begin{cases} 1 \leq i \leq C_1 \\ 1 \leq j \leq C_2 \end{cases}, \\ - S_{aa'}^3 &= \left(G''(x_a^i, x_{a'}^j, u_a^j, u_{a'}^j) \right)_{1 \leq i, j \leq C_2}. \end{aligned}$$

On appelle alors S^1 (resp. S^2, S^3) la matrice formée des blocs $S_{aa'}^1$ (resp. $S_{aa'}^2, S_{aa'}^3$) et S la matrice $\begin{pmatrix} S^1 & S^2 \\ (S^2)^T & S^3 \end{pmatrix}$.

$$\forall a, \quad \alpha_a = \begin{pmatrix} \alpha_a^1 \\ \vdots \\ \alpha_a^{K_a} \end{pmatrix} \quad \beta_a = \begin{pmatrix} \beta_a^1 \\ \vdots \\ \beta_a^{K'_a} \end{pmatrix} \quad p_a = \begin{pmatrix} p_a^1 \\ \vdots \\ p_a^{K_a} \end{pmatrix} \quad u_a = \begin{pmatrix} u_a^1 \\ \vdots \\ u_a^{K'_a} \end{pmatrix}$$

$$\forall b, \quad c_b = \begin{pmatrix} c_b^1 \\ \vdots \\ c_b^{N_b} \end{pmatrix} \quad t_b = \begin{pmatrix} t_b^1 \\ \vdots \\ t_b^{N'_b} \end{pmatrix}$$

On définit encore les matrices et vecteurs suivants :

$$A = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_K \\ \beta_1 \\ \vdots \\ \beta_K \end{pmatrix} \quad Y = \begin{pmatrix} c_1 \\ \vdots \\ c_N \\ t_1 \\ \vdots \\ t_N \end{pmatrix} \quad X = \begin{pmatrix} p_1 & I_d \\ \vdots & \vdots \\ p_K & I_d \\ u_1 & 0_d \\ \vdots & \vdots \\ u_K & 0_d \end{pmatrix}$$

$$S_I = \begin{pmatrix} S & 0 \\ 0 & I_{d(d+1)} \end{pmatrix} \quad S_X = (S \ X) \quad \tilde{A} = \begin{pmatrix} A \\ \gamma_{11} \\ \vdots \\ \gamma_{1d} \\ \vdots \\ \gamma_{d1} \\ \vdots \\ \gamma_{dd} \\ \Delta \end{pmatrix}$$

On note ensuite :

- Φ (resp. Ψ) la matrice ayant pour blocs $M_{ab}P_{ab}^{ij}\Phi_{ab}^{ij}$ (resp. $M_{ab}Q_{ab}^{kl}\Psi_{ab}^{kl}$),
- Φ_1 (resp. Ψ_1) la matrice ayant pour blocs diagonaux $\sum_{bj} M_{ab}P_{ab}^{ij}\Phi_{ab}^{ij}$ (resp. $\sum_{bl} M_{ab}P_{ab}^{kl}\Psi_{ab}^{kl}$),
- Φ_2 (resp. Ψ_2) la matrice ayant pour blocs diagonaux $\sum_{ai} M_{ab}P_{ab}^{ij}\Phi_{ab}^{ij}$ (resp. $\sum_{ak} M_{ab}P_{ab}^{kl}\Psi_{ab}^{kl}$).

On note enfin Ξ (resp. Ξ_1, Ξ_2) la matrice de blocs diagonaux Φ and Ψ (resp. Φ_1 and Ψ_1, Φ_2 and Ψ_2). Ainsi l'énergie s'écrit :

$$E = \tilde{A}^T S_I \tilde{A} + \tilde{A}^T S_X^T \Xi_1 S_X \tilde{A} - 2\tilde{A}^T S_X^T \Xi Y + Y^T \Xi_2 Y$$

Et la minimisation de E donne :

$$\tilde{A} = (S_I + S_X^T \Xi_1 S_X)^{-1} S_X^T \Xi Y$$

5.4.7 Algorithme de minimisation

On spécialise l'algorithme 4 à la mise en correspondance d'objets définis par des points et des vecteurs (cf. Alg. 6).

Algorithm 6 Mise en correspondance d'objets définis par des points et des vecteurs

```

Calculer la matrice de coefficients de splines  $S$ 
Initialiser le vecteur  $A$ 
repeat
  for ( $i := 0; i < N_{iter}; i++$ ) do
    Soft Assign local : mettre à jour les matrices de sous-correspondance  $P_{ab}$ 
    and  $Q_{ab}$ 
    Soft Assign global : mettre à jour la matrice d'appariement  $M$ 
    Déformation : mettre à jour le vecteur de déformation  $A$ 
  end for
  Diminuer la température  $\theta$ 
until  $\theta < \theta_{min}$ 

```

5.5 Applications

Dans toutes les applications, l'opérateur G sera un noyau gaussien.

5.5.1 Mise en correspondance de segments

Il existe de nombreux techniques de mise en correspondance de segments mais la plupart sont traitées dans le cas de mise en correspondance stéréoscopique [63, 49], ce qui n'est pas exactement le cadre qui nous intéresse ici puisque nos déformations sont modélisées par des difféomorphismes et non des homographies. Les homographies constituant un groupe, il serait néanmoins possible de traiter ce cas avec la méthodologie décrite dans 2.

Paramètres du problème

On suppose ici que $\{S_a\}$ et $\{T_b\}$ sont deux ensembles de segments. On note p_a^i , $i = 1, 2$ (resp. q_b^i) les extrémités du segment S_a (resp. T_b), x_a^k , $k = 1, \dots, N_{int}$ (resp. y_b^k , $k = 1, \dots, N_{int}$) des points interpolant le segment S_a (resp. T_b) et u_a^j (resp. v_b^j) le vecteur unitaire tangent au segment S_a (resp. T_b).

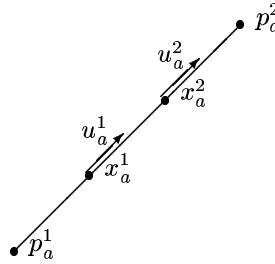
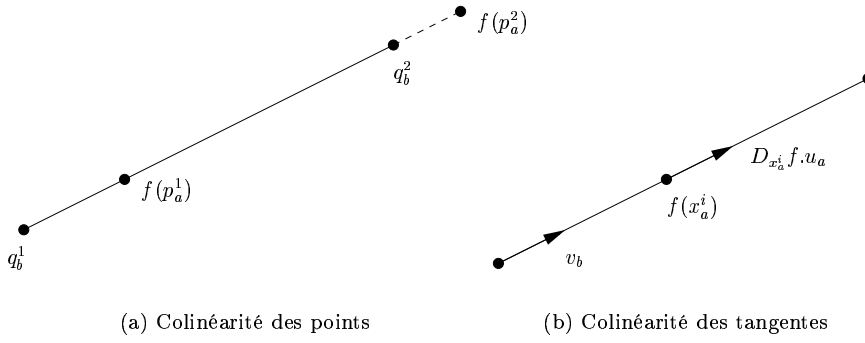


FIG. 5.10 – Segment S_a et ses tangentes aux points interpolants ($N_{int} = 2$)

On a donc dans ce cas $C_1 = 2$ et $C_2 = N_{int}$.

Contraintes

Dans nombre d'applications pratiques, les extrémités de segments sont mal définies. C'est pour cette raison qu'on n'imposera pas l'appariement exact des



(a) Colinéarité des points

(b) Colinéarité des tangentes

extrémités d'un segment S_a avec les extrémités du segment T_b correspondant mais plutôt la *colinéarité* (cf. fig. 5.11(a)).

Si l'on apparie par exemple l'extrémité p_a^i du segment S_a avec l'extrémité q_b^j du segment T_b , cela signifie que l'on veut : $\|(f(p_a^i) - q_b^j) \wedge v_b\|^2 \approx 0$. Avec les notations adoptés dans 5.4, cela signifie :

- $\forall a, b, i, j, \phi_{ab}^{ij} = \|v_b \wedge \cdot\|^2 / \sigma_1^2$,
- $c_b^j = q_b^j$.

On souhaite également que les vecteurs tangents déformés soient parallèles aux vecteurs tangents correspondants (cf. fig. 5.11(b)). Ceci est équivalent à dire que $\|(d_{x_a^k} f \cdot u_a) \wedge v_b\|^2 \approx 0$ pour tout k si le segment S_a est mis en correspondance avec le segment T_b . Avec les notations de 5.4, cela signifie :

- $\forall a, b, k, l, \psi_{ab}^{kl} = \|v_b \wedge \cdot\|^2 / \sigma_2^2$,
- $t_b^l = 0$.

Résultats

On considère un ensemble de N segments qu'on translate chacun d'un vecteur aléatoire et aux extrémités desquels on ajoute un bruit gaussien, ce qui nous procure un second ensemble. Ensuite on ajoute n_1 segments parasites aléatoires dans le premier ensemble et n_2 dans le second.

On applique d'abord notre algorithme avec $N_{int} = 0$, ce qui est équivalent à une mise en correspondance de paires de points. Pour la mise en correspondance de segments véritable, on pose $N_{int} = 1$.

La figure 5.12 donne le résultat des appariements de segments.

Maintenant nous allons appliquer notre algorithme à la mise en correspondance de modèles ("template matching"). Le premier ensemble consiste en un modèle défini par ses arêtes. Le second ensemble comprend plusieurs objets dont le modèle translaté, dilaté, tourné et déformé (bruit gaussien sur les extrémités des segments). On veut donc appairer le modèle avec l'objet correspondant dans le second ensemble. La figure 5.13 montre que notre algorithme aboutit au bon résultat à la différence de la mise en correspondance de points.

L'algorithme travaille en 3D aussi bien qu'en 2D. Voici un exemple : le modèle est une sorte de bâtiment (cf. fig. 5.14) et la cible est une déformation du modèle (rotation, translation, dilatation et addition d'un bruit gaussien sur les extrémités de segments) à laquelle on ajoute plusieurs segments aléatoires. Dans l'exemple de la fig. 5.14, l'angle de rotation est $\pi/8$, le vecteur de translation est

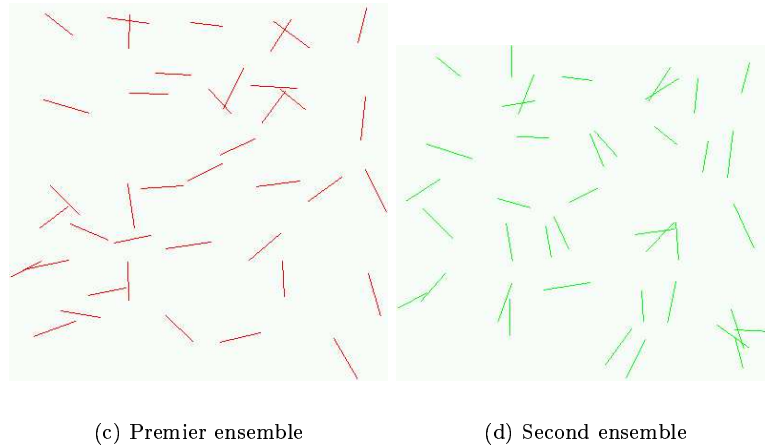
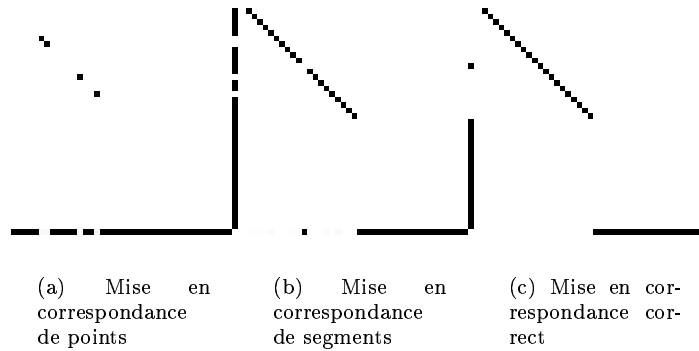
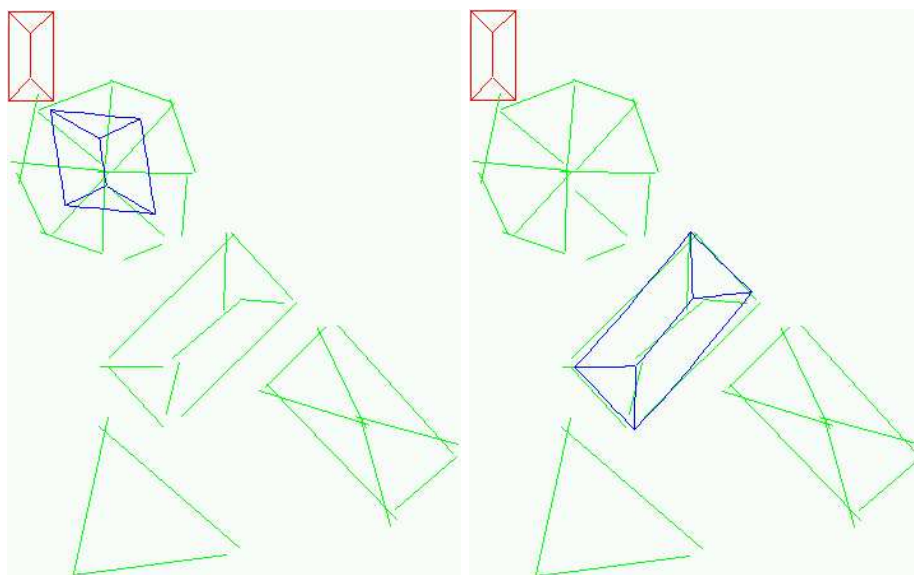
FIG. 5.11 – Deux ensembles de segments avec $N = 20$, $n_1 = n_2 = 20$ 

FIG. 5.12 – Appariement des segments trouvée par les deux algorithmes. Plus le pixel est foncé, plus la probabilité d'appariement est élevée. La dernière ligne et la dernière colonne représentent les outliers.

de l'ordre de la taille du modèle, le facteur de dilatation est 1, 2 et l'écart-type du bruit gaussien est de 5% de la taille du modèle. On a encore ajouté 30 segments aléatoires (le modèle en comporte 17). Comme nous connaissons l'appariement exact, il est possible de calculer le minimum théorique de l'énergie, qui est proche de l'énergie calculée par l'algorithme de minimisation (cf. fig. 5.14).

On présente enfin une expérience sur des données réelles. Le but est de mettre en correspondance des segments extraits de deux images stéréoscopiques. L'algorithme d'extraction de segments est le suivant : une transformée de Hough permet d'extraire les droites caractéristiques de l'image puis on extrait les portions de droite correspondant aux forts gradients dans la direction normale à la droite. On peut voir le résultat de ces extractions dans les figures 5.15 et 5.16 où les segments portant le même numéro sont mis en correspondance.



(a) Mise en correspondance de points

(b) Mise en correspondance de segments

FIG. 5.13 – Mise en correspondance du modèle par les deux algorithmes. L'objet rouge est le modèle, les verts sont la cible et le bleu est le modèle ayant subi la déformation trouvée par l'algorithme. Le nombre N_{int} de tangentes est encore une fois fixé à 1.

5.5.2 Mise en correspondance de lignes

On teste notre algorithme sur la mise en correspondance de lignes (cf. fig 5.17). Ceci n'est qu'une extension de la mise en correspondance de segments. Chaque trait d'un idéogramme est modélisé par 5 points et 4 vecteurs tangents pris entre ces points. On se donne un modèle (fig. 5.17(a)) qu'on essaie de retrouver dans un ensemble d'idéogrammes (fig. 5.17(b)). Le modèle comporte 6 traits et la cible comporte 52 traits. La figure 5.17(c) montre la position relative des deux ensembles lorsqu'ils sont introduits dans l'algorithme et la figure 5.17(d) montre en bleu le modèle déformé prouvant ainsi le succès de la recherche du modèle dans la cible.

5.5.3 Mise en correspondance de polygones

On décrit un polygone par ses sommets ainsi que les tangentes au milieu de ces côtés. Les contraintes utilisés sont l'égalité des sommets et le parallélisme des tangentes. On utilise notre algorithme pour comparer des silhouettes de bâtiments saisies à la main sur des MNE. La figure 5.18 donne le résultat de la mise en correspondance. Malgré la présence d'outliers dans l'ensemble de départ et dans l'ensemble d'arrivée et l'approximation différente des contours dans chacun des ensembles, on constate que l'ensemble d'arrivée et l'ensemble de départ déformé se superposent bien dans leurs parties communes.

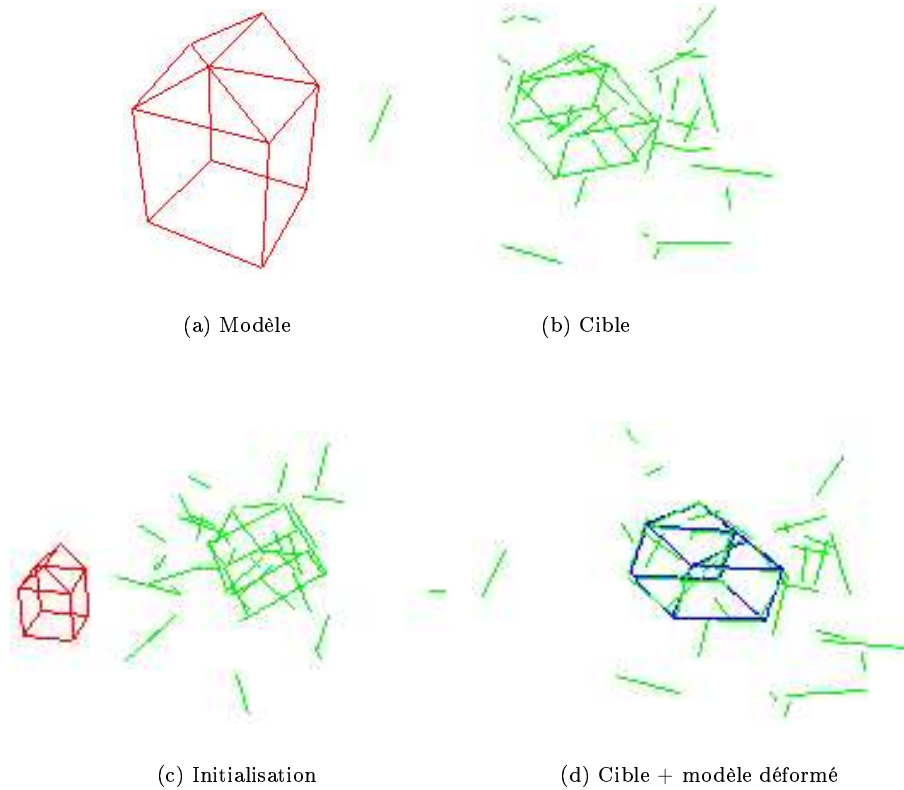


FIG. 5.14 – Retrouver un bâtiment à partir d'un modèle

5.5.4 Mise en correspondance de facettes

On peut aussi définir une facette 3D par des points et un vecteur normal unitaire (cf. 5.19). Les contraintes sont la coplanarité des points des sommets déplacés avec la facette correspondante et le parallélisme des vecteurs normaux déformés avec les vecteurs normaux correspondants.

On note :

- p_a^i (resp. q_b^j) les sommets de la facette S_a (resp. T_b),
- x_a le centre de la facette S_a ,
- u_a (resp. v_b) le vecteur normal à la facette S_a (resp. T_b).

Si l'on veut reprendre les notations de 5.4, cela donne :

- $\phi_{ab}^{ij} = (v_b \cdot \cdot)^2 / \sigma_1^2$,
- $\psi_{ab}^{kl} = \|v_b \wedge \cdot\|^2 / \sigma_2^2$,
- $c_b^j = q_b^j$,
- $t_b^l = 0$.

Voici un nouvel exemple de mise en correspondance de modèle (cf. fig. 5.20). On se donne plusieurs objets géométriques simples décrits par des facettes bruitées (bruit gaussien sur les sommets) et on tente de retrouver un modèle dans cette scène 3D.

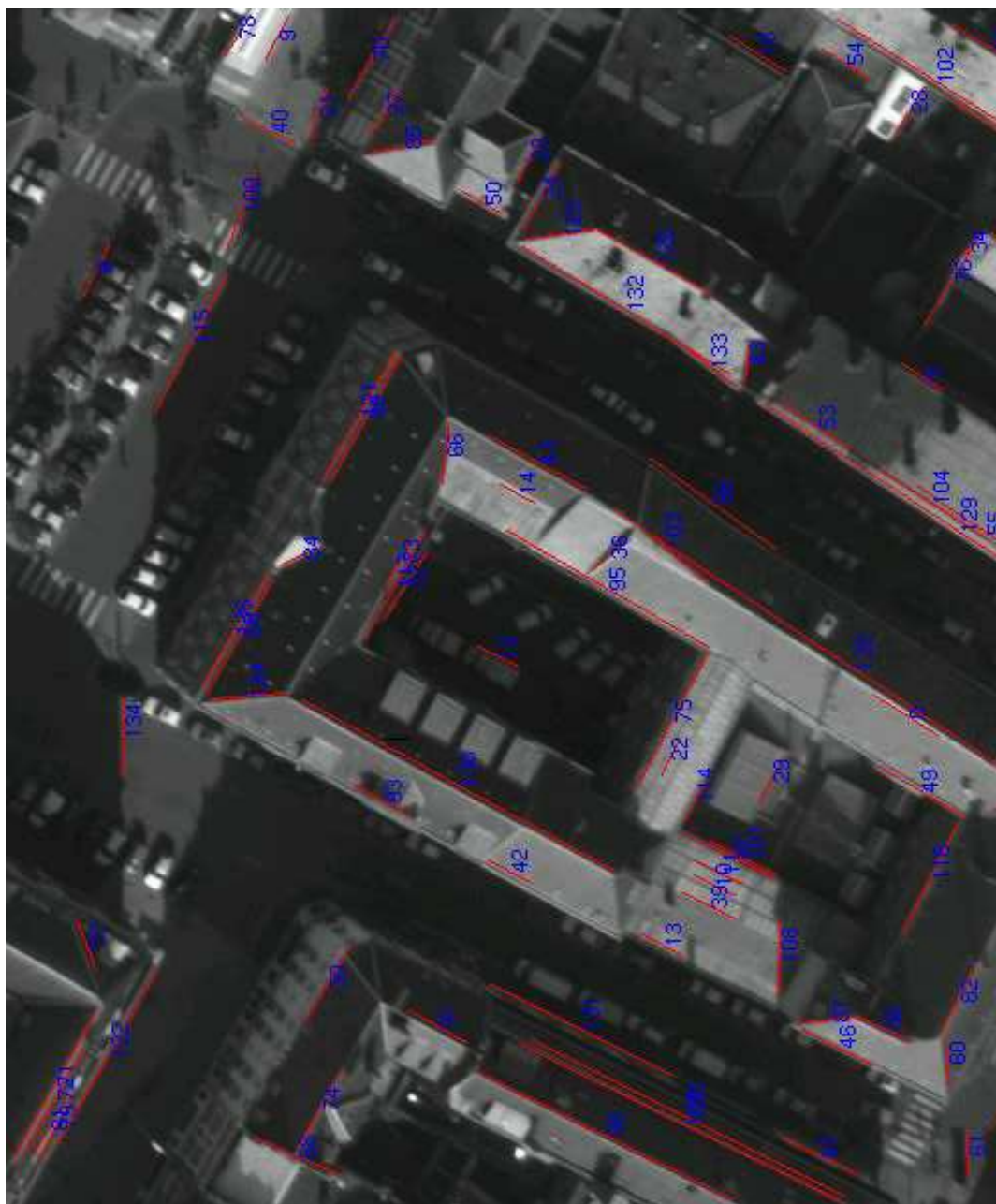


FIG. 5.15 – Première image stéréoscopique. Ensemble de segments de départ

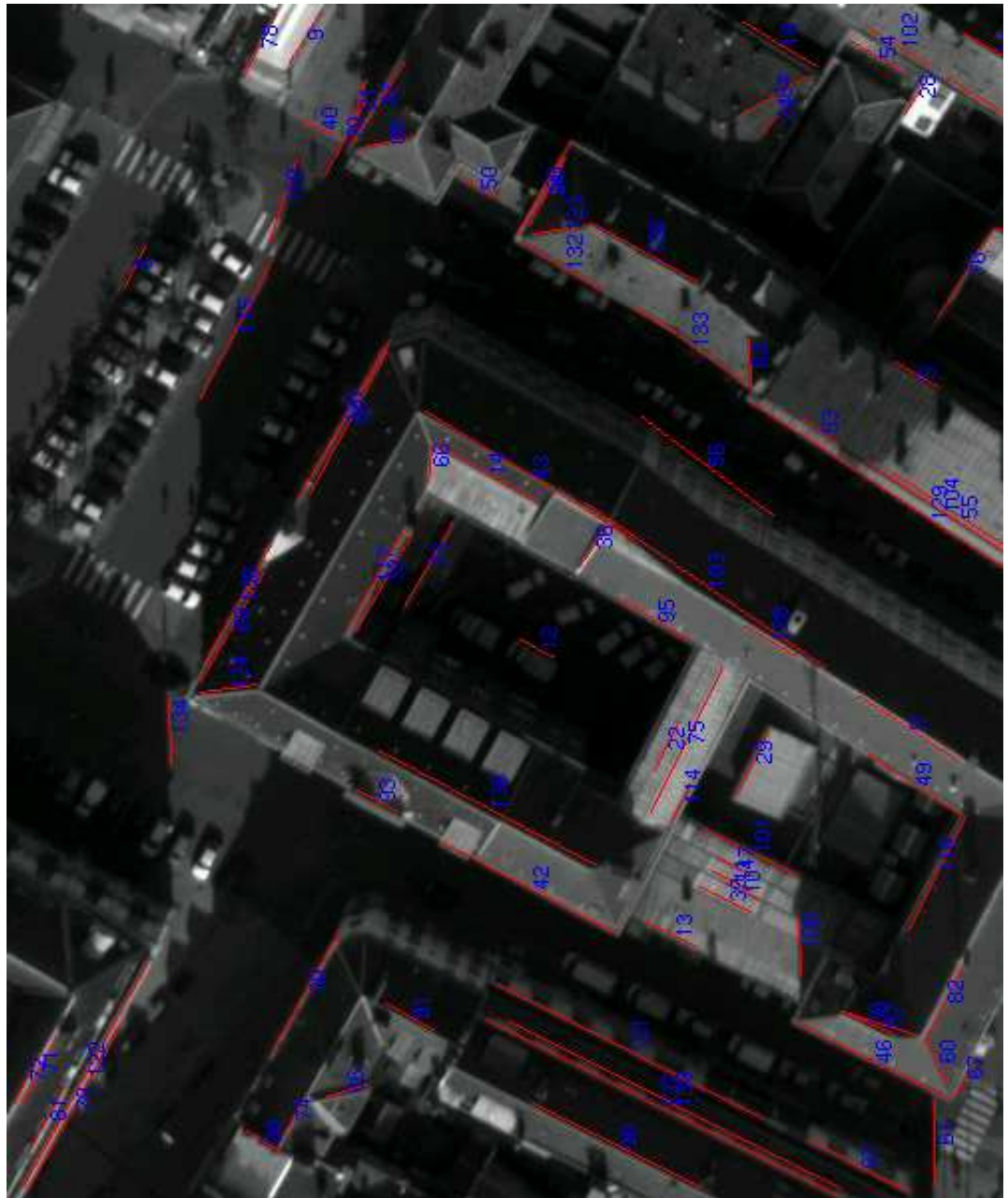


FIG. 5.16 – Seconde image stéréoscopique. Ensemble de segments d'arrivée.

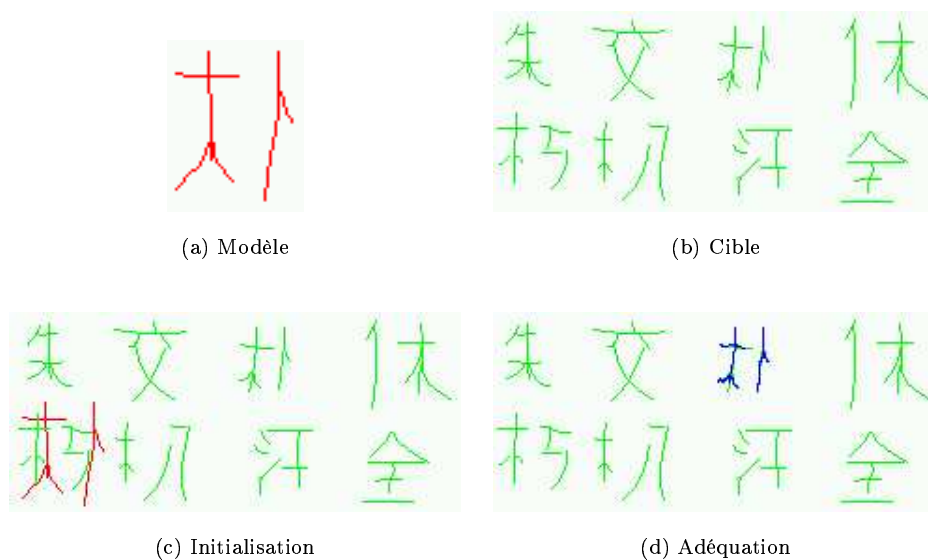


FIG. 5.17 – Mise en correspondance de lignes

5.6 Conclusion

Dans ce chapitre, on a introduit un cadre théorique permettant la mise en correspondance d'objets variés et la mise en oeuvre de contraintes diverses. Le problème de l'appariement nous a empêchés de nous placer dans le cadre des grandes déformations. On aurait cependant pu utiliser le cadre géodésique en mettant en correspondance "en bloc" toutes les composantes des derniers niveaux hiérarchiques mais on aurait alors perdu la notion de hiérarchie des ensembles introduite au début du chapitre.

Il reste aussi à tester la méthode sur des objets comportant plus de deux niveaux hiérarchiques. Il serait notamment intéressant de voir si l'on peut élaborer des stratégies permettant d'accélérer l'algorithme en tirant partie de la définition hiérarchique des objets.

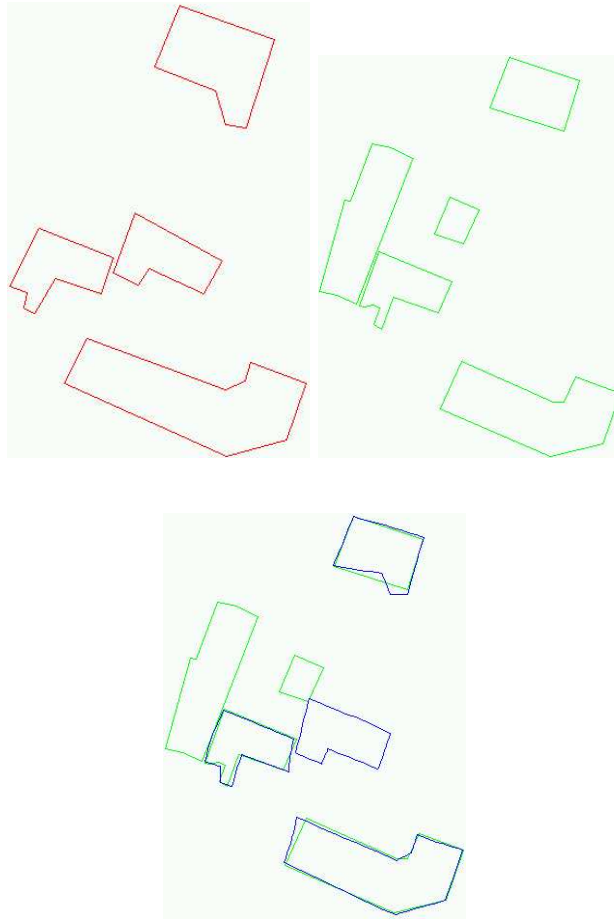


FIG. 5.18 – Mise en correspondance de polygones
Rouge : contour 1, vert : contour 2, bleu : contour 1 déformé.

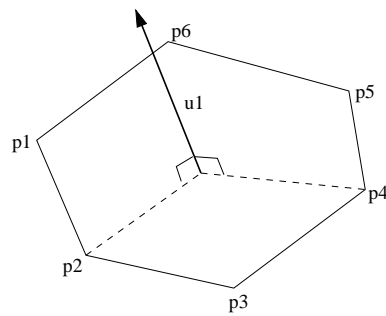


FIG. 5.19 – Une facette définie par ses sommets et son vecteur normal

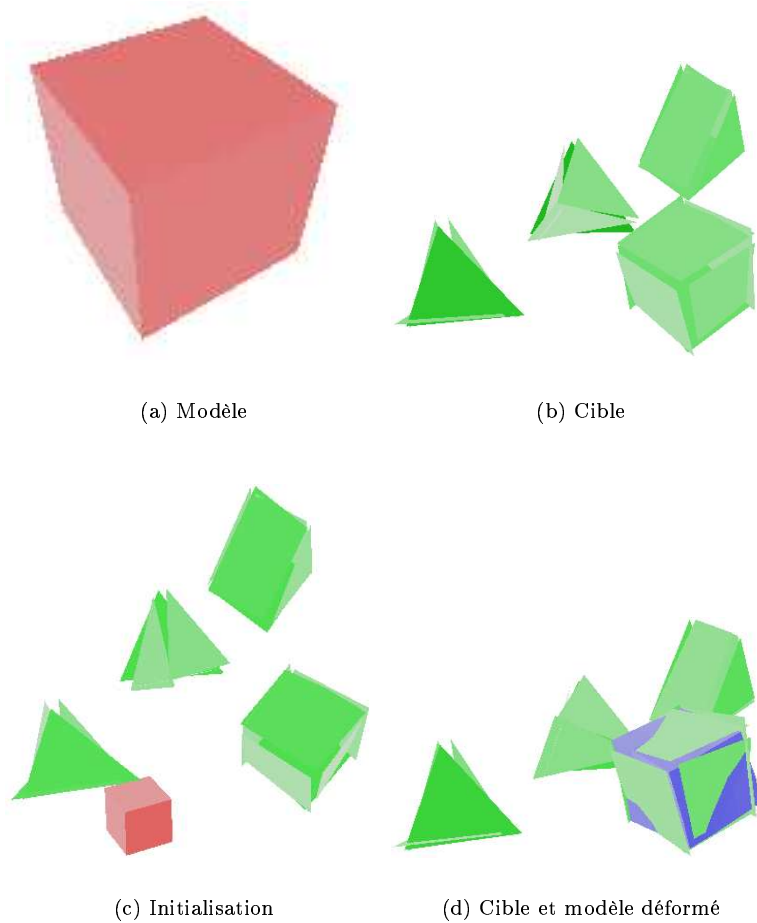


FIG. 5.20 – Retrouver le bon objet 3D

Le cube rouge est le modèle, les objets verts sont la cible et le bleu est le modèle ayant subi la déformation calculée par l'algorithme de minimisation.

Chapitre 6

Mise en correspondance géodésique de primitives

Dans ce chapitre, contrairement au chapitre 5, les correspondances entre primitives sont supposées connues. Il n'y a plus d'appariement à trouver, seulement à résoudre le problème de l'interpolation optimale de la déformation permettant de passer d'un ensemble de points et de vecteurs (tangentes ou normales) à un second ensemble du même type. On reprendra l'application définie dans 2.4.2 que l'on étendra à la mise en correspondance de vecteurs.

6.1 Mise en correspondance géodésique de vecteurs

On pose $\mathcal{I} = (\Omega \times \mathcal{R}^d)^n$ i.e. les n-uplets de couples points-vecteurs. On définit l'action de G sur \mathcal{I} de la manière suivante. A tout élément $I = (p_1, u_1, \dots, p_n, u_n)$ de \mathcal{I} , on associe l'élément suivant :

$$g.I = (g^{-1}(p_1), d_{p_1} g^{-1}.u_1, \dots, g^{-1}(p_n), d_{p_n} g^{-1}.u_n)$$

Nous allons maintenant reprendre le même raisonnement que dans 2.4.2 pour obtenir une expression de la distance géodésique. Pour simplifier l'exposé, on considérera un chemin $\mathbf{a} = (\mathbf{g}, \mathbf{p}, \mathbf{u})$ avec un seul couple point-vecteur ($n = 1$). Le raisonnement reste identique pour $n > 1$. On a alors une relation similaire à la relation (2.26) :

$$\begin{aligned} d_{\mathbf{a}(t_0)} R_{g^{-1}} \frac{d\mathbf{a}}{dt}(t_0) &= \left(\frac{\partial \mathbf{g}}{\partial t}(t_0, \mathbf{g}^{-1}(t_0, \cdot)), d_{\mathbf{p}(t_0)} g \cdot \frac{d\mathbf{p}}{dt}, \dots \right. \\ &\quad \left. \dots d_{\mathbf{p}(t)}^2 g \left(\mathbf{u}(t), \frac{d\mathbf{p}}{dt} \right) + d_{\mathbf{p}(t)} g \cdot \frac{d\mathbf{u}}{dt} \right) \end{aligned} \quad (6.1)$$

avec $\mathbf{a}(t_0) = (g, I)$.

On fait à nouveau le même type de changement de variable que dans (2.27)

et (2.28) :

$$\mathbf{v}(t, x) = \frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{g}^{-1}(t, x)) \quad (6.2)$$

$$\mathbf{q}(t) = \mathbf{g}(t, \mathbf{p}(t)) \quad (6.3)$$

$$\mathbf{w}(t) = d_{\mathbf{p}(t)}\mathbf{g}(t) \cdot \mathbf{u}(t) \quad (6.4)$$

Pour rester cohérent avec les notations du chapitre 2, on peut noter $\mathbf{J}(t) = (\mathbf{q}(t), \mathbf{w}(t))$.

On peut alors écrire :

$$\begin{aligned} \frac{d\mathbf{q}}{dt} &= d_{\mathbf{p}(t)}\mathbf{g}(t) \cdot \frac{d\mathbf{p}}{dt} + \frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{p}(t)) \\ &= d_{\mathbf{p}(t)}\mathbf{g}(t) \cdot \frac{d\mathbf{p}}{dt} + \mathbf{v}(t, \mathbf{q}(t)) \\ \frac{d\mathbf{w}}{dt} &= d_{\mathbf{p}(t)}\mathbf{g}(t) \cdot \frac{d\mathbf{u}}{dt} + d_{\mathbf{p}(t)}^2\mathbf{g}(t) \left(\mathbf{u}(t), \frac{d\mathbf{p}}{dt} \right) + d_{\mathbf{p}(t)}\frac{\partial \mathbf{g}}{\partial t} \cdot \mathbf{u}(t) \\ &= d_{\mathbf{p}(t)}\mathbf{g}(t) \cdot \frac{d\mathbf{u}}{dt} + d_{\mathbf{p}(t)}^2\mathbf{g}(t) \left(\mathbf{u}(t), \frac{d\mathbf{p}}{dt} \right) + d_{\mathbf{q}(t)}\mathbf{v} \cdot \mathbf{w}(t) \end{aligned}$$

On définit à les familles de normes N_G et N_I comme dans l'hypothèse 2.12 mais on va diviser N_I dans le cas présent entre une partie concernant les points et une partie concernant les vecteurs.

Hypothèse 6.1. Soit $\alpha = (v, \delta) = (v, \mathbf{p}, \mathbf{u}) \in T_{Id, I}$ avec $I = (p, u)$. On définit la norme sur $T_{Id, I}$ comme suit :

$$\|\alpha\|_{Id, I}^2 = N_G(v, I)^2 + N_I(i; I)^2 = N_G(v; I)^2 + N_I^0(\mathbf{p}; p)^2 + N_I^1(\mathbf{u}; u)^2 \quad (6.5)$$

Ainsi on peut exprimer l'énergie infinitésimale en t_0 d'un chemin \mathbf{a} tel que $\mathbf{a}(t_0) = (g, I)$:

$$\begin{aligned} \left\| \frac{d\mathbf{a}}{dt}(t_0) \right\|_{g, I}^2 &= \left\| d_{\mathbf{a}(t_0)}R_{g^{-1}} \cdot \frac{d\mathbf{a}}{dt}(t_0) \right\|_{Id, g^{-1} \cdot I}^2 \\ &= N_G(\mathbf{v}(t_0, \cdot); \mathbf{J}(t_0, \cdot))^2 \\ &\quad + N_I^0 \left(\frac{d\mathbf{q}}{dt}(t_0) - \mathbf{v}(t_0, \mathbf{q}(t_0)); \mathbf{q}(t_0) \right)^2 \\ &\quad + N_I^1 \left(\frac{d\mathbf{w}}{dt}(t_0) - d_{\mathbf{q}(t_0)}\mathbf{v}(t_0) \cdot \mathbf{w}(t_0); \mathbf{w}(t_0) \right)^2 \end{aligned}$$

L'expression de l'énergie du chemin est alors :

$$\begin{aligned} E(\mathbf{v}, \mathbf{q}, \mathbf{w}) &= \int_0^1 N_G(\mathbf{v}(t); \mathbf{J}(t))^2 dt \\ &\quad + \int_0^1 N_I^0 \left(\frac{d\mathbf{q}}{dt} - \mathbf{v}(t, \mathbf{q}(t)); \mathbf{q}(t) \right)^2 dt \\ &\quad + \int_0^1 N_I^1 \left(\frac{d\mathbf{w}}{dt} - d_{\mathbf{q}(t)}\mathbf{v}(t) \cdot \mathbf{w}(t); \mathbf{w}(t) \right)^2 dt \end{aligned}$$

Dans nos applications, on ne fera pas dépendre les normes des variables \mathbf{q} et \mathbf{w} . On considèrera que le champ de vitesse $\mathbf{v}(t)$ appartient à un certain espace de Hilbert à noyau reproduisant \mathcal{V} de sorte que $N_G(\cdot) = \|\cdot\|_{\mathcal{V}}$ et on prendra tout simplement la norme L^2 pour N_T^0 et N_T^1 . Ainsi l'expression finale de la distance géodésique est dans ce cas et pour un nombre de couples points-vecteurs quelconque :

$$E(\mathbf{v}, \mathbf{q}, \mathbf{w}) = \int_0^1 \int_{\Omega} \|\mathbf{v}(t)\|_{\mathcal{V}}^2 dx dt + \sum_{i=1}^n \int_0^1 \left\| \frac{d\mathbf{w}_i}{dt} - d_{\mathbf{q}_i(t)} \mathbf{v}(t) \cdot \mathbf{w}_i(t) \right\|^2 dt \quad (6.6)$$

A l'instar de l'exemple présenté dans 2.4.2, la deuxième partie de l'énergie mesure l'adéquation entre les trajectoires des vecteurs et le champ de vitesse \mathbf{v} en comparant la "vitesse" d'un vecteur \mathbf{w}_i placé en un point \mathbf{q}_i à ce même vecteur déformé par le champ \mathbf{v} (ou plus précisément par la différentielle du champ au point \mathbf{q}_i).

6.2 Mise en correspondance géodésique de vecteurs unitaires

Dans la pratique, on a plutôt à mettre en correspondance des directions plutôt que réellement des vecteurs. C'est pourquoi on va s'intéresser à la mise en correspondance de vecteurs unitaires qui sont adaptés pour représenter des directions sans notion de longueur.

On considère ici $\mathcal{I} = (\Omega \times \mathcal{S}^{d-1})^n$ l'ensemble des n-uplets de couples points-vecteurs unitaires. On définit l'action de G sur \mathcal{I} autrement que dans 6.1. A tout élément $I = (p_1, u_1, \dots, p_n, u_n)$ de \mathcal{I} , on associe l'élément suivant :

$$g.I = \left(g^{-1}(p_1), \frac{d_{p_1} g^{-1} \cdot u_1}{\|d_{p_1} g^{-1} \cdot u_1\|}, \dots, g^{-1}(p_n), \frac{d_{p_n} g^{-1} \cdot u_n}{\|d_{p_n} g^{-1} \cdot u_n\|} \right)$$

On reprend le même raisonnement que dans la section précédente pour déterminer la distance géodésique. Par souci de simplification, on traite à nouveau le cas d'un chemin $\mathbf{a} = (\mathbf{g}, \mathbf{p}, \mathbf{u})$ avec un seul couple point-vecteur ($n = 1$). Le calcul de la différentielle de l'action de groupe donne un résultat différent de (2.26) :

$$\begin{aligned} d_{\mathbf{a}(t_0)} R_{g^{-1}} \frac{d\mathbf{a}}{dt}(t_0) &= \left(-\frac{\partial \mathbf{g}^{-1}}{\partial t}(t_0, \mathbf{g}(t_0, \cdot)), d_{\mathbf{p}(t)} g \cdot \frac{d\mathbf{p}}{dt}, \dots \right. \\ &\quad \left. \dots \frac{\hat{\mathbf{u}}(t)}{\|\hat{\mathbf{u}}(t)\|} - \frac{\tilde{\mathbf{u}}(t)}{\|\tilde{\mathbf{u}}(t)\|} \left\langle \frac{\tilde{\mathbf{u}}(t)}{\|\tilde{\mathbf{u}}(t)\|}, \frac{\hat{\mathbf{u}}(t)}{\|\hat{\mathbf{u}}(t)\|} \right\rangle \right) \end{aligned} \quad (6.7)$$

où le chemin \mathbf{a} est tel que $\mathbf{a}(t_0) = (g, I)$ et où l'on définit les variables auxiliaires suivantes :

$$\tilde{\mathbf{u}}(t) = d_{\mathbf{p}(t)} g \cdot \mathbf{u}(t) \quad (6.8)$$

$$\hat{\mathbf{u}}(t) = d_{\mathbf{p}(t)}^2 g \left(\mathbf{u}(t), \frac{d\mathbf{p}}{dt} \right) + d_{\mathbf{p}(t)} g \cdot \frac{d\mathbf{u}}{dt} \quad (6.9)$$

On fait à nouveau le même type de changement de variable que dans (6.2), (6.3) et (6.4) :

$$\mathbf{v}(t, x) = \frac{\partial \mathbf{g}}{\partial t}(t, \mathbf{g}^{-1}(t, x)) \quad (6.10)$$

$$\mathbf{q}(t) = \mathbf{g}(t, \mathbf{p}(t)) \quad (6.11)$$

$$\mathbf{w}(t) = \frac{d_{\mathbf{p}(t)} \mathbf{g}(t) \cdot \mathbf{u}(t)}{\|d_{\mathbf{p}(t)} \mathbf{g}(t) \cdot \mathbf{u}(t)\|} \quad (6.12)$$

Pour rester cohérent avec les notations du chapitre 2, on note à nouveau $\mathbf{J}(t) = (\mathbf{q}(t), \mathbf{w}(t))$.

On peut alors écrire :

$$\begin{aligned} \frac{d\mathbf{q}}{dt} &= d_{\mathbf{p}(t)} \mathbf{g}(t) \cdot \frac{d\mathbf{p}}{dt} + \frac{\partial \mathbf{g}^{-1}}{\partial t}(t, \mathbf{p}(t)) \\ &= d_{\mathbf{p}(t)} \mathbf{g}(t) \cdot \frac{d\mathbf{p}}{dt} + \mathbf{v}(t, \mathbf{q}(t)) \\ \frac{d\mathbf{w}}{dt} &= \frac{\hat{\mathbf{u}}(t) + d_{\mathbf{p}(t)} \frac{\partial \mathbf{g}}{\partial t} \cdot \mathbf{u}(t)}{\|\hat{\mathbf{u}}(t)\|} - \frac{\tilde{\mathbf{u}}(t)}{\|\tilde{\mathbf{u}}(t)\|} \left\langle \frac{\tilde{\mathbf{u}}(t)}{\|\tilde{\mathbf{u}}(t)\|}, \frac{\hat{\mathbf{u}}(t) + d_{\mathbf{p}(t)} \frac{\partial \mathbf{g}}{\partial t} \cdot \mathbf{u}(t)}{\|\hat{\mathbf{u}}(t)\|} \right\rangle \\ &= \frac{\hat{\mathbf{u}}(t)}{\|\hat{\mathbf{u}}(t)\|} + d_{\mathbf{q}(t)} \mathbf{v}(t) \cdot \mathbf{u}(t) - \mathbf{w}(t) \left\langle \mathbf{w}(t), \frac{\hat{\mathbf{u}}(t)}{\|\hat{\mathbf{u}}(t)\|} + d_{\mathbf{q}(t)} \mathbf{v}(t) \cdot \mathbf{u}(t) \right\rangle \end{aligned}$$

On définit à nouveau les normes N_G et N_I à l'aide de l'hypothèse 6.1.

Ainsi on peut écrire l'énergie infinitésimale en t_0 d'un chemin \mathbf{a} tel que $\mathbf{a}(t_0) = (g, I)$:

$$\begin{aligned} \left\| \frac{d\mathbf{a}}{dt}(t_0) \right\|_{g, I}^2 &= \left\| d_{\mathbf{a}(t_0)} R_{g^{-1}} \cdot \frac{d\mathbf{a}}{dt}(t_0) \right\|_{I, d, g^{-1} \cdot I}^2 \\ &= N_G(\mathbf{v}(t_0, \cdot); \mathbf{J}(t_0, \cdot))^2 \\ &\quad + N_I^0 \left(\frac{d\mathbf{q}}{dt}(t_0) - \mathbf{v}(t_0, \mathbf{q}(t_0)); \mathbf{q}(t_0) \right)^2 \\ &\quad + N_I^1 \left(\frac{d\mathbf{w}}{dt}(t_0) - d_{\mathbf{q}(t_0)} \mathbf{v}(t_0) \cdot \mathbf{w}(t_0) + \mathbf{w}(t_0) \langle \mathbf{w}(t_0), d_{\mathbf{q}(t_0)} \mathbf{v}(t_0) \cdot \mathbf{w}(t_0) \rangle; \mathbf{w}(t_0) \right)^2 \end{aligned}$$

L'expression de l'énergie du chemin est alors :

$$\begin{aligned} E(\mathbf{v}, \mathbf{q}, \mathbf{w}) &= \int_0^1 N_G(\mathbf{v}(t); \mathbf{J}(t))^2 dt \\ &\quad + \int_0^1 N_I^0 \left(\frac{d\mathbf{q}}{dt} - \mathbf{v}(t, \mathbf{q}(t)); \mathbf{q}(t) \right)^2 dt \\ &\quad + \int_0^1 N_I^1 \left(\frac{d\mathbf{w}}{dt} - d_{\mathbf{q}(t)} \mathbf{v}(t) \cdot \mathbf{w}(t) + \mathbf{w}(t) \langle \mathbf{w}(t), d_{\mathbf{q}(t)} \mathbf{v}(t) \cdot \mathbf{w}(t) \rangle; \mathbf{w}(t) \right)^2 dt \end{aligned}$$

On prend à nouveau $N_G(\cdot) = \|\cdot\|_{\mathcal{V}}$ où \mathcal{V} est un espace de Hilbert à noyau reproduisant et $N_I^0 = N_I^1 = L^2$. Ainsi l'expression finale de la distance géodésique

est dans ce cas et pour un nombre de couples points-vecteurs quelconque :

$$\begin{aligned}
E(\mathbf{v}, \mathbf{q}, \mathbf{w}) &= \int_0^1 \int_{\Omega} \|\mathbf{v}(t)\|_{\mathcal{V}}^2 dx dt \\
&+ \sum_{i=1}^n \int_0^1 \left\| \frac{d\mathbf{w}_i}{dt} - d_{\mathbf{q}_i(t)} \mathbf{v}(t) \cdot \mathbf{w}_i(t) + \mathbf{w}_i(t) \langle \mathbf{w}_i(t), d_{\mathbf{q}_i(t)} \mathbf{v}(t) \cdot \mathbf{w}_i(t) \rangle \right\|^2 dt
\end{aligned} \tag{6.13}$$

L'expression finale de l'énergie (6.13) diffère donc de celle de la section précédente (6.6). En effet, on considère ici des vecteurs unitaires dont les dérivées temporelles leur sont nécessairement normales. Ce qui justifie que le terme $d_{\mathbf{q}_i(t)} \mathbf{v}(t) \cdot \mathbf{w}_i(t)$ dans (6.6) soit maintenant remplacé par sa projection sur l'hyperplan orthogonal à $\mathbf{w}_i(t)$.

6.3 Formalisation par les splines

On expose pour l'instant les formules pour la mise en correspondance de vecteurs quelconques. On expliquera plus bas (cf. section 6.5) comment adapter la méthode à la mise en correspondance de vecteurs unitaires.

On peut maintenant mêler la mise en correspondance de points à la mise en correspondance de vecteurs. Si l'on note maintenant \mathbf{p}_i les trajectoires des points, et $(\mathbf{q}_j, \mathbf{w}_j)$ les trajectoires des couples points-vecteurs, la nouvelle énergie de mise en correspondance peut s'écrire :

$$\begin{aligned}
E(\mathbf{v}, \mathbf{p}, \mathbf{q}, \mathbf{w}) &= \int_0^1 \int_{\Omega} \|\mathbf{v}(t, \cdot)\|_{\mathcal{V}}^2 dx dt \\
&+ \sum_i \int_0^1 \left\| \frac{d\mathbf{p}_i}{dt} - \mathbf{v}(t, \mathbf{p}_i(t)) \right\|^2 dt \\
&+ \sum_j \int_0^1 \left\| \frac{d\mathbf{w}_j}{dt} - d_{\mathbf{q}_j(t)} \mathbf{v}(t) \cdot \mathbf{w}_j(t) \right\|^2 dt
\end{aligned} \tag{6.14}$$

On remarque que l'énergie (6.14) est du même type que celle dans (4.7) avec une intégration sur le temps. On va donc pouvoir la minimiser de la même façon.

Le champ de vitesse minimisant l'énergie est donc à nouveau de la forme (4.10) :

$$\mathbf{v}(t, \cdot) = \sum_i K(\mathbf{p}_i(t), \cdot) \cdot \alpha_i(t) + \sum_j (\partial_1 K(\mathbf{q}_j(t), \cdot) \cdot \mathbf{w}_j(t)) \cdot \beta_j(t) \tag{6.15}$$

où K est le noyau reproduisant de \mathcal{V} et les α_i et les β_j sont des vecteurs.

Dans cette application aux splines géodésiques, les formes quadratiques ϕ_i et ψ_j de 4.2.3 sont toutes égales à l'identité. Nous allons donc émettre une hypothèse sur le noyau K qui nous permettra de minimiser l'énergie sur chaque composante de h indépendamment.

Hypothèse 6.2. *On suppose que K est un noyau à valeurs scalaires ou, pour être plus correct, $K = k \cdot I_d$ où k est un noyau scalaire (par la suite, nous identifierons K et k).*

Dans le cas général, K est un noyau à valeurs *matricielles*. On va voir que cette hypothèse nous permet aussi d'exprimer l'énergie de façon simple.

6.4 Algorithme de minimisation de l'énergie

On définit les matrices suivantes :

- $F_1(t)$ est la matrice de termes $K(\mathbf{p}_i(t), \mathbf{p}_j(t))$,
- $F_2(t)$ est la matrice de termes $\partial_1 K(\mathbf{q}_i(t), \mathbf{p}_j(t)) \cdot \mathbf{w}_i(t)$,
- $F_3(t)$ est la matrice de termes $\partial_1 \partial_2 K(\mathbf{q}_i(t), \mathbf{q}_j(t)) (\mathbf{w}_i(t), \mathbf{w}_j(t))$.

On notera aussi l'ensemble des trajectoires sous forme matricielle :

$$P(t) = \begin{pmatrix} \mathbf{p}_1^1(t) & \dots & \mathbf{p}_1^d(t) \\ \vdots & & \vdots \\ \mathbf{p}_n^1(t) & \dots & \mathbf{p}_n^d(t) \end{pmatrix} \quad A(t) = \begin{pmatrix} \alpha_1^1(t) & \dots & \alpha_1^d(t) \\ \vdots & & \vdots \\ \alpha_n^1(t) & \dots & \alpha_n^d(t) \end{pmatrix}$$

$$W(t) = \begin{pmatrix} \mathbf{w}_1^1(t) & \dots & \mathbf{w}_1^d(t) \\ \vdots & & \vdots \\ \mathbf{w}_{n'}^1(t) & \dots & \mathbf{w}_{n'}^d(t) \end{pmatrix} \quad B(t) = \begin{pmatrix} \beta_1^1(t) & \dots & \beta_1^d(t) \\ \vdots & & \vdots \\ \beta_n^1(t) & \dots & \beta_n^d(t) \end{pmatrix}$$

On peut alors donner une expression matricielle de l'énergie :

$$\begin{aligned} E(P, Q, W, A, B) &= \int_0^1 \text{Tr} (A(t)^T F_1(t) A(t) + 2A(t)^T F_2(t) B(t) + B(t)^T F_3(t) B(t)) dt \\ &+ \frac{1}{\sigma_1^2} \int_0^1 \left\| \frac{dP}{dt} - F_1(t) A(t) - F_2(t) B(t) \right\|^2 dt \\ &+ \frac{1}{\sigma_2^2} \int_0^1 \left\| \frac{dW}{dt} - F_2(t)^T A(t) - F_3(t) B(t) \right\|^2 dt \end{aligned} \quad (6.16)$$

On ne peut pas minimiser l'énergie en tous ses paramètres en même temps, il faudra donc minimiser d'abord l'énergie en la déformation puis en les trajectoires et itérer ces deux étapes que nous allons détailler.

6.4.1 Minimisation sur le champ de déformation

C'est un problème classique de moindres carrés. Pour le résoudre, on adopte les notations suivantes :

$$\tilde{P}(t) = \begin{pmatrix} P(t)/\sigma_1 \\ W(t)/\sigma_2 \end{pmatrix} \quad \tilde{A}(t) = \begin{pmatrix} A(t) \\ B(t) \end{pmatrix} \quad (6.17)$$

$$F(t) = \begin{pmatrix} F_1(t) & F_2(t) \\ F_2(t)^T & F_3(t) \end{pmatrix} \quad \tilde{F}(t) = \begin{pmatrix} F_1(t)/\sigma_1 & F_2(t)/\sigma_1 \\ F_2(t)^T/\sigma_2 & F_3(t)/\sigma_2 \end{pmatrix} \quad (6.18)$$

L'énergie peut alors se réécrire de la manière suivante :

$$E(\tilde{P}, \tilde{A}, F, \tilde{F}) = \int_0^1 \text{Tr} (\tilde{A}(t)^T F(t) \tilde{A}(t)) dt + \int_0^1 \left\| \tilde{P}'(t) - \tilde{F}(t) \tilde{A}(t) \right\|^2 dt \quad (6.19)$$

La minimisation de l'énergie sous cette forme donne :

$$\tilde{A}(t) = \left(F(t) + \tilde{F}(t)^T \tilde{F}(t) \right)^{-1} \tilde{F}(t)^T \tilde{P}'(t) \quad (6.20)$$

6.4.2 Minimisation sur les trajectoires

Dans cette partie, on notera :

$$\begin{aligned} K_1(x, y) &= K(x, y) \\ K_2(x, y, u) &= \langle \partial_2 K(x, y), u \rangle \\ K_3(x, y, u, v) &= \partial_1 \partial_2 K(x, y)(u, v) \end{aligned}$$

On procède par une descente de gradient. Il faut donc calculer le gradient de E en les \mathbf{p}_k , \mathbf{q}_k et \mathbf{w}_k . On notera :

$$\begin{aligned} Z_k^1(t) &= \frac{d\mathbf{p}_k}{dt} - \sum_{i=1}^n K_1(\mathbf{p}_k(t), \mathbf{p}_i(t))\alpha_i(t) \\ &\quad - \sum_{j=1}^{n'} K_2(\mathbf{p}_k(t), \mathbf{q}_j(t), \mathbf{w}_j(t))\beta_j(t) \\ Z_k^2(t) &= \frac{d\mathbf{w}_k}{dt} - \sum_{i=1}^n K_2(\mathbf{p}_i(t), \mathbf{q}_k(t), \mathbf{w}_k(t))\alpha_i(t) \\ &\quad - \sum_{j=1}^{n'} K_3(\mathbf{q}_k(t), \mathbf{q}_j(t), \mathbf{w}_k(t), \mathbf{w}_j(t))\beta_j(t) \end{aligned}$$

De cette façon, l'énergie à minimiser s'écrit :

$$\begin{aligned} E(\alpha, \beta, \mathbf{p}, \mathbf{q}, \mathbf{w}) &= \int_0^1 \sum_{k,l} \langle \alpha_k(t), \alpha_l(t) \rangle K_1(\mathbf{p}_k(t), \mathbf{p}_l(t)) dt \\ &\quad + 2 \int_0^1 \sum_{k,l} \langle \alpha_k(t), \beta_l(t) \rangle K_2(\mathbf{p}_k(t), \mathbf{q}_l(t), \mathbf{w}_l(t)) dt \\ &\quad + \int_0^1 \sum_{k,l} \langle \beta_k(t), \beta_l(t) \rangle K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)) dt \\ &\quad + \frac{1}{\sigma_1^2} \int_0^1 \sum_k \|Z_k^1(t)\|^2 dt + \frac{1}{\sigma_2^2} \int_0^1 \sum_k \|Z_k^2(t)\|^2 dt \end{aligned}$$

On suppose que l'étape de minimisation sur les trajectoires succède toujours à l'étape de calcul de la déformation. Dans ce cas, on peut vérifier que l'on a toujours $Z_k^1 = \sigma_1^2 \alpha_k$ et $Z_k^2 = \sigma_2^2 \beta_k$ pour tout k . Ceci nous permettra de simplifier les expressions des gradients de l'énergie en \mathbf{p}_k , \mathbf{q}_k et \mathbf{w}_k .

Pour cela, on va calculer la variation de l'énergie E quand on fait varier la variable \mathbf{p} (resp. \mathbf{q} , \mathbf{w}) de \mathbf{p} (resp. \mathbf{q} , \mathbf{w}) à $\mathbf{p} + \epsilon \mathbf{p}$ (resp. $\mathbf{q} + \epsilon \mathbf{q}$, $\mathbf{w} + \epsilon \mathbf{w}$). Il est essentiel de noter que, les extrémités du chemin \mathbf{p} (resp. \mathbf{q} , \mathbf{w}) étant fixes, on a les contraintes suivantes : $\mathbf{p}(0) = \mathbf{p}(1) = 0$ (resp. $\mathbf{q}(0) = \mathbf{q}(1) = 0$, $\mathbf{w}(0) = \mathbf{w}(1) = 0$).

$$\begin{aligned}
 \frac{1}{2} \frac{d}{d\epsilon} E(\mathbf{p} + \epsilon \mathbf{p}) &= \int_0^1 \sum_{k,l} \langle \partial_1 K_1(\mathbf{p}_k(t), \mathbf{p}_l(t)), \mathbf{p}_k(t) \rangle \langle \alpha_k(t), \alpha_l(t) \rangle dt \\
 &+ \int_0^1 \sum_{k,l} \langle \partial_1 K_2(\mathbf{p}_k(t), \mathbf{q}_l(t), \mathbf{w}_l(t)), \mathbf{p}_k(t) \rangle \langle \alpha_k(t), \beta_l(t) \rangle dt \\
 &+ \frac{1}{\sigma_1^2} \int_0^1 \sum_k \left\langle Z_k^1(t), \frac{d\mathbf{p}_k}{dt} \right\rangle dt \\
 &- \frac{1}{\sigma_1^2} \int_0^1 \sum_{k,l} \langle \partial_1 K_1(\mathbf{p}_k(t), \mathbf{p}_l(t)), \mathbf{p}_k(t) \rangle \langle Z_k^1(t), \alpha_l(t) \rangle dt \\
 &- \frac{1}{\sigma_1^2} \int_0^1 \sum_{k,l} \langle \partial_1 K_1(\mathbf{p}_l(t), \mathbf{p}_k(t)), \mathbf{p}_l(t) \rangle \langle Z_k^1(t), \alpha_l(t) \rangle dt \\
 &- \frac{1}{\sigma_1^2} \int_0^1 \sum_{k,l} \langle \partial_1 K_2(\mathbf{p}_k(t), \mathbf{q}_l(t), \mathbf{w}_l(t)), \mathbf{p}_k(t) \rangle \langle Z_k^1(t), \beta_l(t) \rangle dt \\
 &- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \langle \partial_1 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{p}_l(t) \rangle \langle Z_k^2(t), \alpha_l(t) \rangle dt
 \end{aligned}$$

En utilisant le fait que $Z_k^1 = \sigma_1^2 \alpha_k$ et $Z_k^2 = \sigma_2^2 \beta_k$ ainsi que la symétrie du noyau K et en intégrant par partie les termes en $\frac{d\mathbf{p}_k}{dt}$ l'expression précédente se simplifie en :

$$\begin{aligned}
 \frac{1}{2} \frac{d}{d\epsilon} E(\mathbf{p} + \epsilon \mathbf{p}) &= - \int_0^1 \sum_{k,l} \langle \partial_1 K_1(\mathbf{p}_k(t), \mathbf{p}_l(t)), \mathbf{p}_k(t) \rangle \langle \alpha_k(t), \alpha_l(t) \rangle dt \\
 &- \int_0^1 \sum_{k,l} \langle \partial_1 K_2(\mathbf{p}_k(t), \mathbf{q}_l(t), \mathbf{w}_l(t)), \mathbf{p}_k(t) \rangle \langle \alpha_k(t), \beta_l(t) \rangle dt \\
 &+ \int_0^1 \sum_k \left\langle \frac{d\alpha_k}{dt}, \mathbf{p}_k(t) \right\rangle dt
 \end{aligned}$$

Ce qui donne l'expression du gradient de l'énergie en les \mathbf{p}_k :

$$\begin{aligned}
 \frac{1}{2} \partial_{\mathbf{p}_k} E(t) &= - \sum_{l=1}^n \langle \alpha_k(t), \alpha_l(t) \rangle \partial_1 K_1(\mathbf{p}_k(t), \mathbf{p}_l(t)) \\
 &- \sum_{l=1}^{n'} \langle \alpha_k(t), \beta_l(t) \rangle \partial_1 K_2(\mathbf{p}_k(t), \mathbf{q}_l(t), \mathbf{w}_l(t)) \\
 &- \frac{d\alpha_k}{dt}
 \end{aligned} \tag{6.21}$$

On répète la même opération pour le gradient en \mathbf{q} .

$$\begin{aligned}
\frac{1}{2} \frac{d}{d\epsilon} E(\mathbf{q} + \epsilon \mathbf{q}) &= \int_0^1 \sum_{k,l} \langle \partial_2 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{q}_k(t) \rangle \langle \beta_k(t), \alpha_l(t) \rangle dt \\
&+ \int_0^1 \sum_{k,l} \langle \partial_1 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)), \mathbf{q}_k(t) \rangle \langle \beta_k(t), \beta_l(t) \rangle dt \\
&- \frac{1}{\sigma_1^2} \int_0^1 \sum_{k,l} \langle \partial_2 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{q}_k(t) \rangle \langle Z_l^1(t), \beta_k(t) \rangle dt \\
&- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \partial_2 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t), \mathbf{q}_k(t)) \langle Z_k^2(t), \alpha_l(t) \rangle dt \\
&- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \partial_1 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)), \mathbf{q}_k(t) \rangle \langle Z_k^2(t), \beta_l(t) \rangle dt \\
&- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \partial_1 K_3(\mathbf{q}_l(t), \mathbf{q}_k(t), \mathbf{w}_l(t), \mathbf{w}_k(t)), \mathbf{q}_l(t) \rangle \langle Z_k^2(t), \beta_l(t) \rangle dt
\end{aligned}$$

Ce qui, pour les mêmes raisons que précédemment, se simplifie en :

$$\begin{aligned}
\frac{1}{2} \frac{d}{d\epsilon} E(\mathbf{q} + \epsilon \mathbf{q}) &= - \int_0^1 \sum_{k,l} \langle \partial_2 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{q}_k(t) \rangle \langle \beta_k(t), \alpha_l(t) \rangle dt \\
&- \int_0^1 \sum_{k,l} \langle \partial_1 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)), \mathbf{q}_k(t) \rangle \langle \beta_k(t), \beta_l(t) \rangle dt
\end{aligned}$$

Et nous obtenons le gradient suivant en les \mathbf{q}_k :

$$\begin{aligned}
\frac{1}{2} \partial_{\mathbf{q}_k} E(t) &= - \sum_{l=1}^n \langle \beta_k(t), \alpha_l(t) \rangle \partial_2 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)) \\
&- \sum_{l=1}^{n'} \langle \beta_k(t), \beta_l(t) \rangle \partial_1 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t))
\end{aligned} \tag{6.22}$$

On calcule enfin la variation suivante :

$$\begin{aligned}
 \frac{1}{2} \frac{d}{d\epsilon} E(\mathbf{w} + \epsilon \mathbf{w}) &= \int_0^1 \sum_{k,l} \langle \partial_3 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{w}_k(t) \rangle \langle \alpha_l(t), \beta_k(t) \rangle dt \\
 &+ \int_0^1 \sum_{k,l} \langle \partial_3 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)), \mathbf{w}_k(t) \rangle \langle \beta_k(t), \beta_l(t) \rangle dt \\
 &- \frac{1}{\sigma_1^2} \int_0^1 \sum_{k,l} \langle \partial_3 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{w}_k(t) \rangle \langle Z_l^1(t), \beta_k(t) \rangle dt \\
 &+ \frac{1}{\sigma_2^2} \int_0^1 \sum_k \left\langle Z_k^2(t), \frac{d\mathbf{w}_k}{dt} \right\rangle dt \\
 &- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \langle \partial_3 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)), \mathbf{w}_k(t) \rangle \langle Z_k^2(t), \alpha_l(t) \rangle dt \\
 &- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \langle \partial_3 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)), \mathbf{w}_k(t) \rangle \langle Z_k^2(t), \beta_l(t) \rangle dt \\
 &- \frac{1}{\sigma_2^2} \int_0^1 \sum_{k,l} \langle \partial_3 K_3(\mathbf{q}_l(t), \mathbf{q}_k(t), \mathbf{w}_l(t), \mathbf{w}_k(t)), \mathbf{w}_l(t) \rangle \langle Z_k^2(t), \beta_l(t) \rangle dt
 \end{aligned}$$

Expression qui se simplifie de la manière suivante :

$$\begin{aligned}
 \frac{1}{2} \partial_{\mathbf{w}_k} E(t) &= - \sum_{l=1}^n \langle \beta_k(t), \alpha_l(t) \rangle \partial_3 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)) \\
 &- \sum_{l=1}^{n'} \langle \beta_k(t), \beta_l(t) \rangle \partial_3 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)) \\
 &- \frac{d\beta_k}{dt}
 \end{aligned} \tag{6.23}$$

6.4.3 Schéma algorithmique

On déduit des étapes de mise à jour précédentes l'algorithme de mise en correspondance géodésique (cf. Alg. 7).

Algorithm 7 Mise en correspondance géodésique

Calculer les matrices de spline $F_1(t)$, $F_2(t)$, $F_3(t)$

Initialiser les vecteurs $A(t)$, $B(t)$

repeat

 Mettre à jour la déformation (i.e. les vecteurs $A(t)$, $B(t)$) par moindres carrés

 Mettre à jour les trajectoires \mathbf{p} , \mathbf{q} , \mathbf{w} par descente de gradient (une seule étape)

 Mettre à jour les matrices de spline $F_1(t)$, $F_2(t)$, $F_3(t)$

until $\Delta E < \epsilon$

6.5 Adaptation de la méthode à la mise en correspondance de vecteurs unitaires

6.5.1 Forme du champ de vitesse

Il faut être un peu plus précautionneux pour montrer que le champ de vitesse \mathbf{v} minimisant l'énergie (6.13) doit être de la forme décrite dans (6.15). En effet, l'énergie (6.13) n'est pas du même type que (4.7) en ce qui concerne la partie faisant intervenir la différentielle du champ \mathbf{v} . Néanmoins si l'on note ξ_w la forme quadratique sur \mathbb{R}^d telle que $\xi_w(u) = \|u - w\langle w, u \rangle\|^2$ pour $w \in \mathbb{R}^d$, on remarque que :

$$\begin{aligned} \xi_{\mathbf{w}(t)} \left(\frac{d\mathbf{w}}{dt} - d_{\mathbf{q}(t)}\mathbf{v}(t) \cdot \mathbf{w}(t) \right) \\ = \left\| \frac{d\mathbf{w}}{dt} - d_{\mathbf{q}(t)}\mathbf{v}(t) \cdot \mathbf{w}(t) - \mathbf{w}(t) \left\langle \mathbf{w}(t), \frac{d\mathbf{w}}{dt} \right\rangle + \mathbf{w}(t) \langle \mathbf{w}(t), d_{\mathbf{q}(t)}\mathbf{v}(t) \cdot \mathbf{w}(t) \rangle \right\|^2 \end{aligned}$$

Or $\mathbf{w}(t)$ étant un vecteur unitaire, $\langle \mathbf{w}(t), \frac{d\mathbf{w}}{dt} \rangle = 0$ et on peut donc finalement se placer dans le cadre d'une énergie du type (4.7). Le champ \mathbf{v} est donc bien de la forme (6.15).

6.5.2 Minimisation de l'énergie en la déformation

On ne peut malheureusement plus exprimer l'énergie de la même manière que dans (6.16). En effet, les projections sur les hyperplans $\mathbf{w}_k(t)^\perp$ intervenant dans l'énergie (6.13) nous empêchent de traiter chaque composante indépendamment et on ne peut plus écrire l'énergie sous une forme matricielle analogue à (6.16).

On définit à nouveau les matrices suivantes :

- $F_1(t)$ est la matrice de blocs $K(\mathbf{p}_i(t), \mathbf{p}_j(t))I_d$,
- $F_2(t)$ est la matrice de blocs $\partial_1 K(\mathbf{q}_i(t), \mathbf{p}_j(t)) \cdot \mathbf{w}_i(t) I_d$,
- $F_3(t)$ est la matrice de blocs $\partial_1 \partial_2 K(\mathbf{q}_i(t), \mathbf{q}_j(t)) (\mathbf{w}_i(t), \mathbf{w}_j(t)) I_d$.

On note de plus $P(t)$ (resp. $Q(t)$, $W(t)$, $A(t)$, $B(t)$) le vecteur formé par la concaténation des vecteurs $\mathbf{p}_i(t)$ (resp. $\mathbf{q}_i(t)$, $\mathbf{w}_i(t)$, $\alpha_i(t)$, $\beta_i(t)$). Enfin, on note $\Pi(t)$ la matrice de blocs diagonaux $I_d - \mathbf{w}_i(t)\mathbf{w}_i(t)^T$.

L'énergie peut alors s'écrire :

$$\begin{aligned} E(P, Q, W, A, B) \\ = \int_0^1 (A(t)^T F_1(t) A(t) + 2A(t)^T F_2(t) B(t) + B(t)^T F_3(t) B(t)) dt \\ + \frac{1}{\sigma_1^2} \int_0^1 \left\| \frac{dP}{dt} - F_1(t) A(t) - F_2(t) B(t) \right\|^2 dt \\ + \frac{1}{\sigma_2^2} \int_0^1 \left\| \frac{dW}{dt} - \Pi(t) (F_2(t)^T A(t) + F_3(t) B(t)) \right\|^2 dt \end{aligned} \quad (6.24)$$

La minimisation sur le champ de déformation donne formellement le même résultat que dans 6.4.1 à la différence près que l'on a ici :

$$\tilde{F}(t) = \begin{pmatrix} F_1(t)/\sigma_1 & F_2(t)/\sigma_1 \\ \Pi(t)F_2(t)^T/\sigma_2 & \Pi(t)F_3(t)/\sigma_2 \end{pmatrix}$$

6.5.3 Minimisation de l'énergie en les trajectoires

En ce qui concerne la minimisation de l'énergie sur les trajectoires, on peut vérifier que le calcul des gradients de l'énergie en les \mathbf{p}_k , \mathbf{q}_k donne le même résultat que dans (6.21) et (6.22) (toujours sous l'hypothèse que le gradient est calculé juste après l'étape de déformation). Le gradient de l'énergie en \mathbf{w}_k est légèrement différent :

$$\begin{aligned} \frac{1}{2} \partial_{\mathbf{w}_k} E(t) = & - \sum_{l=1}^n \langle \beta_k(t), \alpha_l(t) \rangle \partial_3 K_2(\mathbf{p}_l(t), \mathbf{q}_k(t), \mathbf{w}_k(t)) \\ & - \sum_{l=1}^{n'} \langle \beta_k(t), \beta_l(t) \rangle \partial_3 K_3(\mathbf{q}_k(t), \mathbf{q}_l(t), \mathbf{w}_k(t), \mathbf{w}_l(t)) \\ & - \frac{d\beta_k}{dt} - \frac{1}{\sigma_2^2} \langle Z_k^2, \mathbf{w}_k \rangle \tilde{Z}_k^2 \end{aligned}$$

avec Z_k^2 défini comme dans 6.4.2 et $\tilde{Z}_k^2 = Z_k^2 - \langle Z_k^2, \mathbf{w}_k \rangle \mathbf{w}_k$.

6.6 Applications

Dans les applications, le noyau K sera un noyau gaussien d'écart-type σ . On utilise l'algorithme pour mettre en correspondance deux contours 2D. Les contours sont en fait des polygones et l'on tente de mettre en correspondance les sommets des polygones ainsi que les vecteurs directeurs des segments composant les polygones pris au milieu des segments. La correspondance entre les points est supposée connue. On utilise une discrétisation en temps de 10 pas.

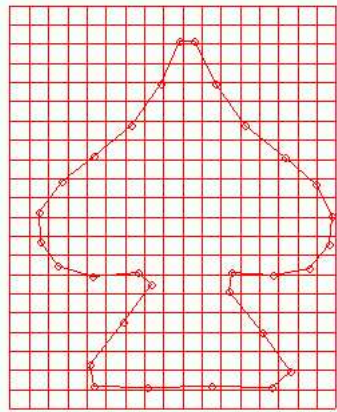
On compare les résultats avec ceux obtenus grâce à des splines classiques. Le cas classique correspond à une discrétisation en temps d'un seul pas dans notre algorithme. Les figures 6.1 et 6.3 montrent bien que les splines géodésiques génèrent un véritable difféomorphisme contrairement aux splines classiques.

Nous avons aussi implémenté la mise en correspondance à base de vecteurs unitaires mais, dans les exemples traités, on a obtenu des résultats quasi identiques à ceux de la mise en correspondance de vecteurs quelconques.

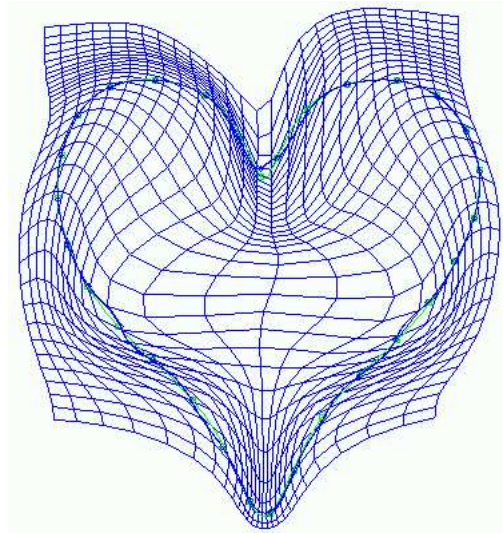
6.7 Perspectives

On n'a pas pu inclure de déformations affines dans le cadre de la mise en correspondance géodésique comme on l'a fait dans 4.2.5. L'algorithme qui serait ainsi généré ne serait pas stable. En effet, on pourrait constater que la minimisation donnerait lieu à une implosion puis une explosion de la déformation à cause du facteur d'échelle de la transformation affine. On aurait néanmoins pu prendre en compte des rotations. Mais il faut être plus rigoureux si l'on veut inclure une partie affine arbitraire dans la déformation. Ceci a été fait dans [55] et pourra s'adapter de manière directe à la mise en correspondance de vecteurs.

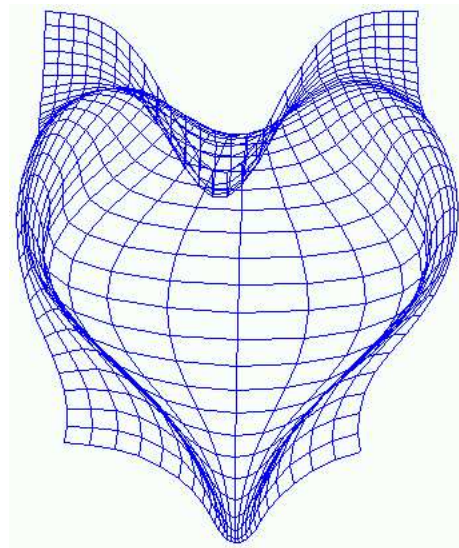
On pourrait également introduire un aspect multi-résolution comme il a été suggéré dans 4.3.



(a) Contour initial



(b) Contour final : splines géodésiques



(c) Contour final : splines classiques

FIG. 6.1 – Mise en correspondance de contours

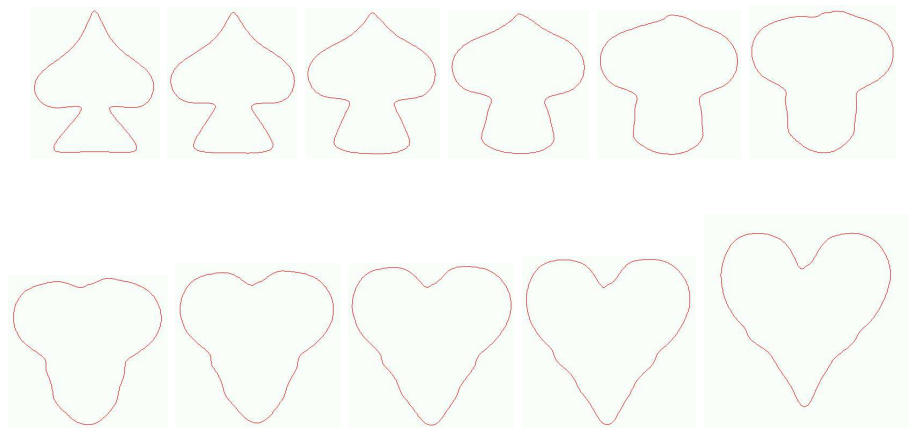
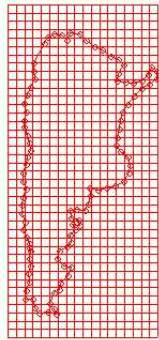
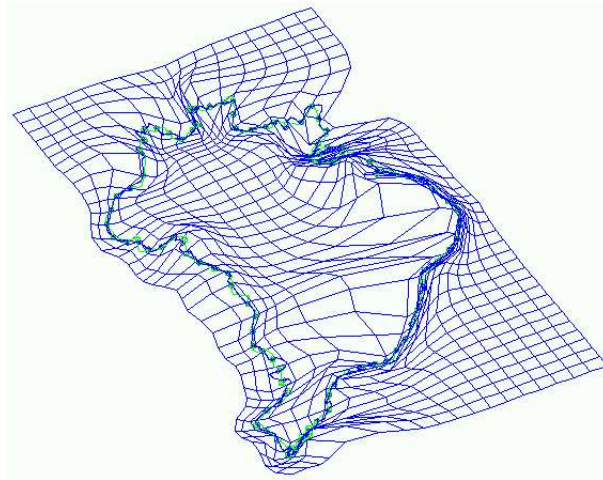


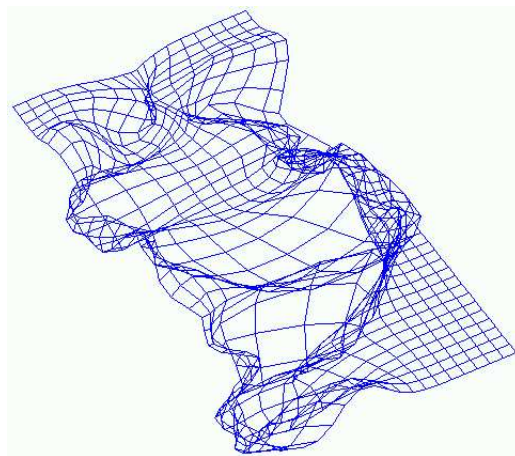
FIG. 6.2 – Evolution du contour au cours du temps



(a)
Contour
initial



(b) Contour final : splines géodésiques



(c) Contour final : splines classiques

FIG. 6.3 – Mise en correspondance de contours

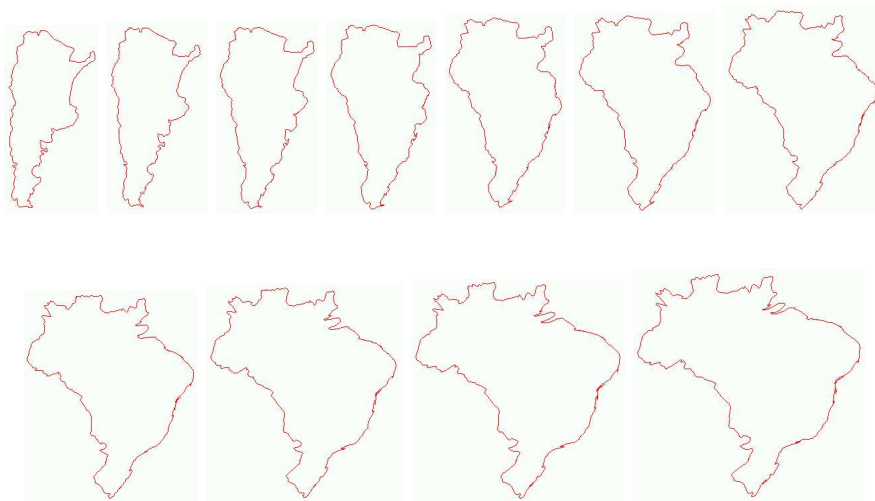


FIG. 6.4 – Evolution du contour au cours du temps

Chapitre 7

Mise en correspondance géodésique de primitives par quantification

7.1 Problématique

Dans le chapitre 5, on a exposé une méthode permettant d'effectuer un appariement et une interpolation de la déformation à la fois tandis que la mise en correspondance géodésique exposé dans le chapitre 6, bien que plus précis, présupposait que l'appariement avait déjà été effectué. On souhaiterait donc inventer une nouvelle méthode permettant de bénéficier des avantages des deux méthodes précédentes.

On précise à l'avance qu'on ne s'intéressera qu'à des primitives définies par des ensembles de points bien qu'une d'extension pour des segments soit ébauchée dans 7.6. Si l'on s'intéresse à des techniques de mise en correspondance de formes on peut par exemple se reporter à [15] où l'on définit des approximations de métriques sur des ensembles de contours et non plus sur des ensembles de N -uplets de points comme on a pu le faire au chapitre précédent.

Plutôt que de mettre directement des points en correspondance, on va diviser les formes à apparier en parties plus petites que nous appellerons des *quanta* et on mettra en correspondance les centres de ces quanta, réduisant par la même la combinatoire du problème de l'appariement. Mieux, on peut carrément s'affranchir de l'étape d'appariement en estimant simultanément les quanta et le difféomorphisme, les deux problèmes se contraignant l'un l'autre. L'idée est encore une fois de définir une énergie de mise en correspondance faisant intervenir à la fois un terme d'adéquation des points avec les quanta, un terme contraignant le difféomorphisme par les centres de quanta et un terme de régularisation du difféomorphisme [27].

7.2 Algorithme de quantification

Bien que la quantification ne soit pas le but de l'algorithme présenté dans ce chapitre, on a non seulement besoin de quanta pour s'affranchir de tout pro-

blème d'appariement mais aussi car leurs centres vont servir de *landmarks* pour définir la déformation. On a donc besoin de s'intéresser de plus près au type de quantification que l'on va adopter et en particulier à l'énergie de quantification associée puisque la méthode employée dans ce chapitre est à nouveau de type énergétique.

On suppose que donc l'on a un ensemble de points $V = \{V_m\}$ dans lequel on veut trouver N quanta A_n (le nombre N de quanta est fixé *a priori*) définis par leurs centres C_n . On dira qu'un point V_m appartient au quantum A_n s'il est suffisamment proche de C_n .

7.2.1 K-means

Cet algorithme [26] est un des algorithmes de quantification les plus répandus. L'appartenance des points aux quanta est modélisée par une matrice binaire M définie de la manière suivante :

$$\begin{aligned} M_{mn} &= 1 \Leftrightarrow V_m \in A_n \\ M_{mn} &= 0 \Leftrightarrow V_m \notin A_n \end{aligned}$$

On considère que chaque point ne peut appartenir qu'à un seul quantum à la fois ce qui peut être signifié de la manière suivante :

$$\forall m, \sum_{n=1}^N M_{mn} = 1 \quad (7.1)$$

L'algorithme de K-means consiste alors à minimiser l'énergie suivante :

$$E(V, C) = \sum_{m,n} M_{mn} \|V_m - C_n\|^2 \quad (7.2)$$

Il existe plusieurs algorithmes permettant de minimiser cette énergie, le plus connu étant sans doute l'algorithme dit de Lloyd [38]. Mais là n'est pas le propos de notre exposé. Nous nous en tiendrons donc pour l'instant à la définition de l'énergie de quantification.

7.2.2 Soft K-means

Cette méthode est assez similaire à la méthode des K-means mais elle est plus robuste que cette dernière. On trouvera de plus amples détails sur cet algorithme dans [12, 13].

La différence avec la méthode précédente est que M n'est plus une matrice binaire. On accorde aux coefficients de M le droit de prendre des valeurs réelles entre 0 et 1. Plus précisément, chaque coefficient M_{mn} de M représentera la probabilité qu'aura le point V_m d'appartenir au quantum A_n :

$$M_{mn} = P(\{V_m \in A_n\})$$

Chaque ligne m de M représente la distribution de la probabilité d'appartenance du point V_m aux quanta $\{A_n\}$ si bien que la contrainte (7.1) tient toujours.

En s'inspirant de la physique, on impose la forme d'une distribution de Gibbs à la probabilité d'appartenance du point V_m aux quanta A_n (cf. [47]) :

$$M_{mn} = \frac{1}{Z(\beta)} \exp(-\beta D(V_m, A_n)) \quad (7.3)$$

où $\beta = \frac{1}{T}$ est l'inverse d'un paramètre de température, $Z(\beta)$ est la fonction de partition du système :

$$Z(\beta) = \sum_n \exp(-\beta D(V_m, A_n))$$

et l'énergie d'un état $V_m \in A_n$ est définie simplement comme suit :

$$D(V_m, A_n) = \|V_m - C_n\|^2$$

La physique statistique nous dit alors que l'énergie libre associée au système est égale à :

$$F_T(V_m, A) = \sum_n M_{mn} D(V_m, A_n) + T \sum_n M_{mn} \log M_{mn}$$

Ainsi en sommant sur tous les m , on obtient une nouvelle forme d'énergie :

$$F_T(V, C) = \sum_{m,n} M_{mn} \|V_m - C_n\|^2 + T \sum_{m,n} M_{mn} \log M_{mn} \quad (7.4)$$

On remarque que, quand la température T tend vers 0, l'énergie $F_T(V, C)$ tend vers $E(V, C)$ (cf. eqn. 7.2).

On inclura donc la minimisation de l'énergie dans une procédure de recuit déterministe, ce qui lui donne une robustesse accrue. La minimisation de l'énergie en M est donnée par l'expression (7.3) et la minimisation en C est directe. L'algorithme de quantification est donc donnée dans Alg. 8.

Algorithm 8 Algorithme de Soft K-means

repeat

Mettre à jour la matrice M

$$M_{mn} = \frac{\exp(-\frac{\|V_m - C_n\|^2}{T})}{\sum_p \exp(-\frac{\|V_m - C_p\|^2}{T})}$$

Mettre à jour les centres de quanta

$$C_n = \sum_{m,n} M_{mn} V_m$$

Diminuer la température T

until $T < T_{min}$

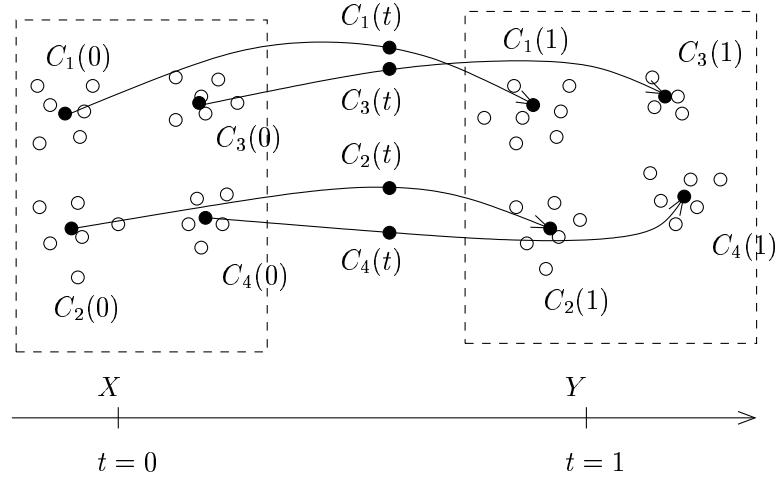


FIG. 7.1 – Les deux ensembles à mettre en correspondance, les centres de quanta et leurs trajectoires

7.3 Mise en correspondance de primitives

On désire maintenant mettre en correspondance deux primitives X et Y définies par des ensembles de points : $X = \{X_1, \dots, X_K\}$ et $Y = \{Y_1, \dots, Y_L\}$. Comme prévu, plutôt que de mettre en correspondance les ensembles de points X et Y , on va apparier des centres de quanta de ces ensembles. On adopte la même méthodologie que dans le chapitre 6, c'est-à-dire que les centres de quanta ont une trajectoire au cours du temps. Au temps $t = 0$, ce sont les centres de quanta de l'ensemble X et au temps $t = 1$, ce sont les centres de quanta de l'ensemble Y . On notera $C = \{C_1, \dots, C_N\}$ l'ensemble des trajectoires des centres de quanta i.e. chaque C_i est une application de $[0; 1]$ dans $\Omega \subset \mathbb{R}^d$. La figure 7.1 résume la situation.

7.3.1 Energie d'adéquation avec les quanta

Les points de X (resp. Y) doivent définir les quanta à l'instant $t = 0$ (resp. $t = 1$). On note M^X (resp. M^Y) la matrice d'appartenance de taille (K, N) (resp. (L, N)) des points de X (resp. Y) aux quanta de $C(0)$ (resp. $C(1)$). On adopte une énergie de quantification du type (7.4).

$$\begin{aligned}
 E_{\text{quanta}}(X, Y, C(0), C(1)) &= E_{\text{quanta}}(X, C(0)) + E_{\text{quanta}}(Y, C(1)) \\
 &= \sum_{kn} M_{kn}^X \|X_k - C_n(0)\|^2 + T \sum_{kn} M_{kn}^X \log M_{kn}^X \\
 &\quad + \sum_{ln} M_{ln}^Y \|Y_l - C_n(1)\|^2 + T \sum_{ln} M_{ln}^Y \log M_{ln}^Y
 \end{aligned} \tag{7.5}$$

7.3.2 Energie liée au difféomorphisme

Le difféomorphisme permet de lier les centres de quanta aux temps $t = 0$ et $t = 1$. On reprend la même énergie que celle définit dans (2.29).

$$E_{diff}(C, v) = \lambda \int_0^1 \|v(t, \cdot)\|_{\mathcal{V}} dt + \sum_n \int_0^1 \left\| \frac{dC_n}{dt} - v(t, C_n(t)) \right\|^2 dt \quad (7.6)$$

où \mathcal{V} est à nouveau un Hilbert à noyau reproduisant $G = g.Id$ et g est un noyau à valeurs scalaires. Il faut noter que, contrairement au chapitre 6, les extrémités des trajectoires ne sont pas fixées. On souhaite d'ailleurs que les extrémités bougent de manière à se placer sur les véritables centres de quanta des ensembles X et Y .

7.4 Minimisation de l'énergie

Il y a trois types de variables en lesquelles on doit minimiser l'énergie :

- les coefficients de la matrice M , autrement dit les probabilités d'appartenance aux quanta,
- les trajectoires C ,
- la déformation v .

Il est difficile de minimiser l'énergie en toutes ces variables en même temps mais il est possible de minimiser l'énergie en chacun de ces trois types de variables séparément. Nous allons donc détailler les trois étapes de minimisation que cela implique.

7.4.1 Etape de quantification

Dans un premier temps on calcule les matrices \hat{M}^X et \hat{M}^Y minimisant l'énergie courante à C et v fixés :

$$\hat{M}_{kn}^X = \exp\left(-\frac{\|X_k - C_n(0)\|^2}{T}\right)$$

$$\hat{M}_{ln}^Y = \exp\left(-\frac{\|Y_l - C_n(1)\|^2}{T}\right)$$

On normalise ensuite les lignes des matrices \hat{M}^X et \hat{M}^Y de telle manière que la somme de leurs éléments soit égale à 1 comme dans 7.2.2.

7.4.2 Calcul de la déformation

Pour les mêmes raisons que dans 6.3, la déformation minimisant l'énergie est de la forme :

$$v(t, \cdot) = \sum_n G(C_n(t), \cdot) \alpha_n(t)$$

où les $\alpha_n(t)$ sont des vecteurs de dimension d . On se reportera à 6.4.1 pour les détails de la minimisation, le cas présent étant plus simple puisqu'il n'intervient que des points et plus de vecteurs.

7.4.3 Calcul des trajectoires

On procède par une descente de gradient. Il convient donc de calculer le gradient de l'énergie en les C_n . Pour cela, on pose $H(\epsilon) = E(C + \epsilon D)$ et on calcule la dérivée suivante :

$$\begin{aligned} \frac{1}{2} \frac{dH}{d\epsilon} &= - \sum_{km} M_{km}^X \langle X_k - C_m(0), D_m(0) \rangle - \sum_{lm} M_{lm}^Y \langle Y_l - C_m(1), D_m(1) \rangle \\ &\quad - \lambda \int_0^1 \sum_{mn} \langle \partial_1 G(C_m(t), C_n(t)), D_m(t) \rangle \langle \alpha_m(t), \alpha_n(t) \rangle dt \\ &\quad + \lambda \int_0^1 \sum_m \left\langle \alpha_m(t), \frac{dD_m}{dt} \right\rangle dt \end{aligned} \quad (7.7)$$

Ce qui donne en intégrant le dernier terme par parties :

$$\begin{aligned} \frac{1}{2} \frac{dH}{d\epsilon} &= - \sum_{km} M_{km}^X \langle X_k - C_m(0), D_m(0) \rangle - \sum_{lm} M_{lm}^Y \langle Y_l - C_m(1), D_m(1) \rangle \\ &\quad - \lambda \int_0^1 \sum_{mn} \langle \partial_1 G(C_m(t), C_n(t)), D_m(t) \rangle \langle \alpha_m(t), \alpha_n(t) \rangle dt \\ &\quad - \lambda \int_0^1 \sum_m \left\langle \frac{d\alpha_m}{dt}, D_m(t) \right\rangle dt + \lambda \sum_m \langle \alpha_m(1), D_m(1) \rangle - \langle \alpha_m(0), D_m(0) \rangle \end{aligned} \quad (7.8)$$

On en tire donc l'expression du gradient de E :

$$\begin{aligned} \frac{1}{2} \partial_{C_m} E(t) &= -\lambda \sum_n \partial_1 G(C_m(t), C_n(t)) \alpha_m(t)^T \alpha_n(t) - \lambda \frac{d\alpha_m}{dt} \\ &\quad - \left(\sum_k M_{km}^X (X_k - C_m(0)) + \lambda \alpha(0) \right) \delta_0(t) \\ &\quad - \left(\sum_l M_{lm}^Y (Y_l - C_m(1)) - \lambda \alpha(1) \right) \delta_1(t) \end{aligned} \quad (7.9)$$

où δ_0 et δ_1 représentent les masses de Dirac en 0 et 1.

7.4.4 Algorithme

Après avoir défini ces deux procédures on peut donner le schéma algorithmique de la minimisation de l'énergie. Néanmoins, il reste une dernière précaution à prendre. En effet, on n'est pas assuré que l'algorithme donne bien des géodésiques, c'est-à-dire s'il ne reste pas bloqué dans un minimum local. Ce pourrait aussi être le cas si les extrémités des chemins C_n étaient fixes mais l'estimation des extrémités des chemins empire les choses comme l'illustre la figure 7.2 en prenant l'exemple des géodésiques de la sphère. On voit que si l'on tire même continûment sur les extrémités de la géodésique dessinée sur la sphère (fig. 7.2(a)), il viendra un moment où le chemin de plus courte longueur

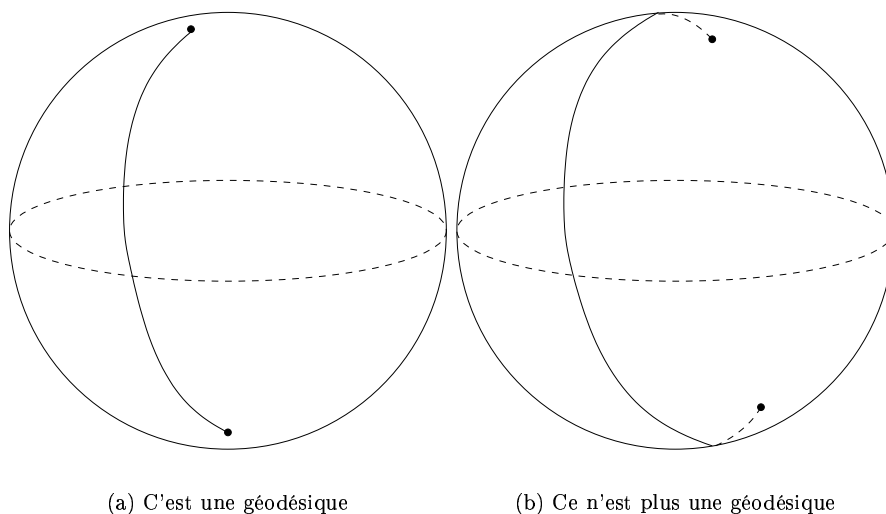


FIG. 7.2 – Passage continu d'une géodésique à une non-géodésique

ne sera plus celui dessiné dans la fig 7.2(b) mais celui passant de l'autre côté de la sphère.

Si on ne prend pas garde à ce problème, on constate en effet que l'algorithme peut donner dans certains cas des résultats peu satisfaisants. Notre parade consiste à tester régulièrement si le chemin en "ligne droite" (i.e. $C(t) = (1-t)C(0) + tC(1)$) ne donne pas une énergie inférieure. Les résultats prouvent que cette solution est acceptable.

L'algorithme final est donné dans Alg. 9.

7.5 Résultats

Dans toutes les applications, le noyau reproduisant sera un noyau gaussien. On remarque qu'on retrouve quasiment le même résultat dans la figure 6.2 et dans la figure 7.5 bien que, dans ce dernier cas, on ne connaisse pas *a priori* les correspondances entre les centres de quanta. On voit aussi que le fait de ne pas tenir compte des tangentes comme dans le chapitre 6 fait que le contour déformé garde les points de rebroussement du contour de départ (fig. 7.4). On voit également à nouveau (fig. 7.6) que les splines classiques ne fournissent pas un difféomorphisme.

On peut aussi comme prévu mettre en correspondance des données $3D$: dans la figure 7.7, on estime la déformation entre deux triangulations de surfaces d'hippocampes¹ fournies par l'Université Johns Hopkins (Center for Imaging Science). Le modèle (en rouge) est définie par une triangulation de 10292 points et la cible (en vert) par une triangulation de 5942 points. On a utilisé 300 quanta et 10 pas de temps dans cette expérience. En bleu, on peut observer le modèle auquel on a appliqué la déformation retrouvée par l'algorithme. Dans la

¹L'hippocampe est une partie du cortex permettant de faire le lien entre la mémoire à court terme et la mémoire à long terme.

Algorithm 9 Quantification et mise en correspondance géodésique

```

Calculer la matrice de spline  $G$ 
Initialiser le vecteur  $A$ 
repeat
  repeat
    Mettre à jour la déformation (i.e. le vecteur  $A$ ) par moindres carrés
    Mettre à jour les trajectoires  $C$  par descente de gradient (une seule étape)
    Calculer l'énergie  $E$ 
    Calculer l'énergie  $E'$  du chemin en "ligne droite"  $C'(t) = (1 - t)C(0) + tC(1)$ 
    if  $E' < E$  then
       $C \leftarrow C'$ 
    end if
    Mettre à jour la matrice de spline  $G$ 
  until  $\Delta E < \epsilon$ 
  Mettre à jour les matrices de quanta  $M^X$  et  $M^Y$ 
  Diminuer la température  $T$ 
until  $T < T_{min}$ 

```

figure 7.8, on peut observer le module de la déformation en chaque point de la surface initiale. On obtient bien la déformation à laquelle on pouvait s'attendre – c'est-à-dire que la "queue" de l'hippocampe se raccourcit et que la déformation y est donc plus importante que sur le reste de la surface.

7.5.1 Consistance

On veut étudier l'influence du nombre de quanta sur les résultats de l'algorithme. Pour cela, on compare les déformations résultant de l'algorithme pour différentes valeurs du nombre de quanta. La figure 7.9 montre que pour des formes définies par 201 points, le résultat évolue peu à partir de 20 quanta.

7.5.2 Robustesse

On étudie ici la robustesse de l'algorithme par rapport au bruit. On rajoute donc un certain nombre de points aléatoires aux ensembles de points définissant les deux formes à mettre en correspondance. Les résultats sont rassemblés dans la figure 7.10 où les ensembles à mettre en correspondance comportent chacun 201 points auxquels on ajoute un nombre N d'outliers. Le nombre de quanta est à chaque fois égal au quart du nombre total de points.

Dans l'expérience suivante (fig. 7.11), les deux ensembles à mettre en correspondance sont encore formés de 201 points mais on ne rajoute des points que dans l'ensemble cible et on prend un nombre de quanta égal à 40 dans chaque cas.

7.6 Perspectives

On souhaiterait être capable d'utiliser autre chose que des points pour définir les objets à mettre en correspondance, en particulier des segments. Le problème

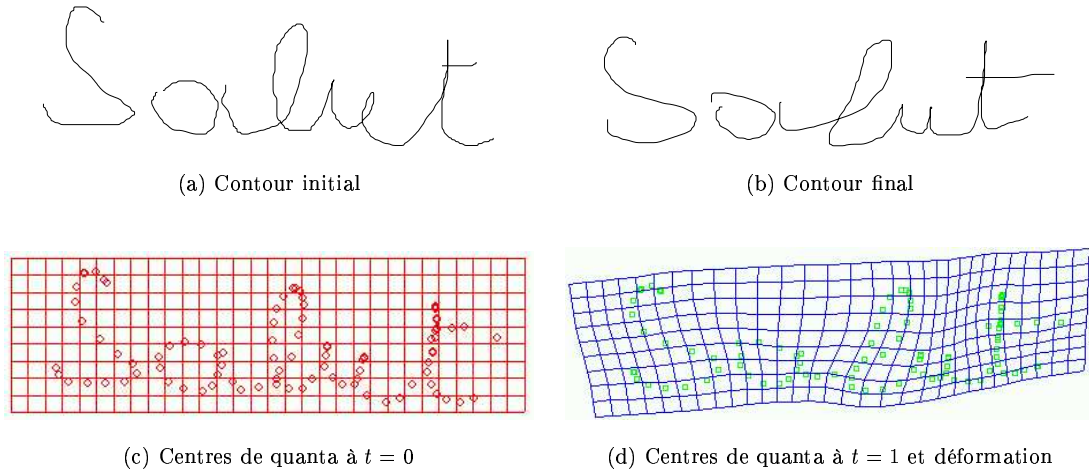


FIG. 7.3 – Mise en correspondance de mots manuscrits

est que l'on ne peut plus définir les quanta par leur centre : en effet, si l'on peut utiliser une moyenne de points pour définir le centre du quantum, cela devient délicat pour des segments par exemple.

Une solution pourrait être de définir un quantum comme un objet différent de ceux qui lui appartiennent, par exemple une ellipse si on travaille en 2D (un ellipsoïde en 3D) comme le schéma de la fig. 7.12. Un segment appartiendrait au quantum A_n si ces extrémités étaient incluses dans l'ellipse associée au quantum. Les quanta A_n seraient donc définies par le centre de l'ellipse ainsi que par une matrice de covariance Σ_n contenant l'information sur l'orientation et la longueur des axes de l'ellipse. En fait, on peut définir ces quanta de manière moins géométrique et un peu plus statistique. On définit donc les quanta A_n comme des lois gaussiennes sur \mathbb{R}^d centrées en C_n et de covariance Σ_n . La probabilité qu'un point X de \mathbb{R}^d appartienne au cluster A_n est alors :

$$P(X \in A_n) = \frac{\exp(-(X - C_n)^T \Sigma_n^{-1} (X - C_n))}{\sqrt{2\pi \text{Det}(\Sigma_n)}}$$

et la "distance" entre X et A_n serait² :

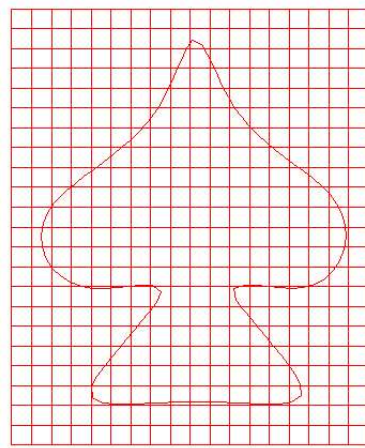
$$D(X, A_n) = -\log(P(X \in A_n)) = (X - C_n)^T \Sigma_n^{-1} (X - C_n) + \frac{1}{2} \log \text{Det}(\Sigma_n)$$

L'énergie de quantification mesurerait alors l'écart entre les extrémités des segments et les différentes lois gaussiennes :

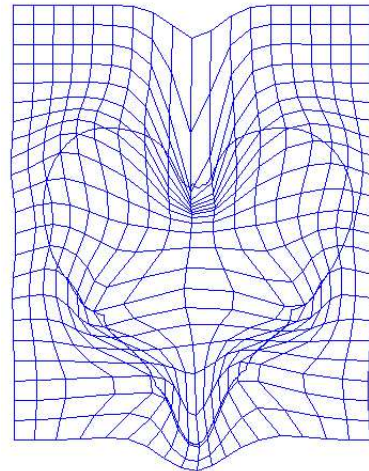
$$E_{\text{quanta}}(X, A) = \sum_{kn} \left[M_{kn} (X_k - C_n)^T \Sigma_n^{-1} (X_k - C_n) + \sum_n \log \text{Det} \Sigma_n \right]$$

où les X_k seraient les extrémités des segments.

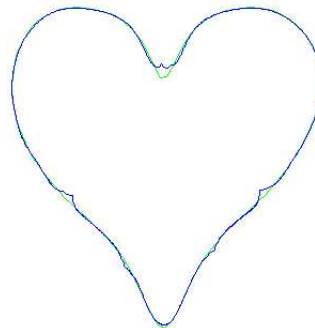
²On néglige la constante en π



(a) Contour initial



(b) Contour final et déformation



(c) Adéquation du contour déformé (bleu) et du contour final (vert)

FIG. 7.4 – Résultat de la mise en correspondance par splines géodésiques

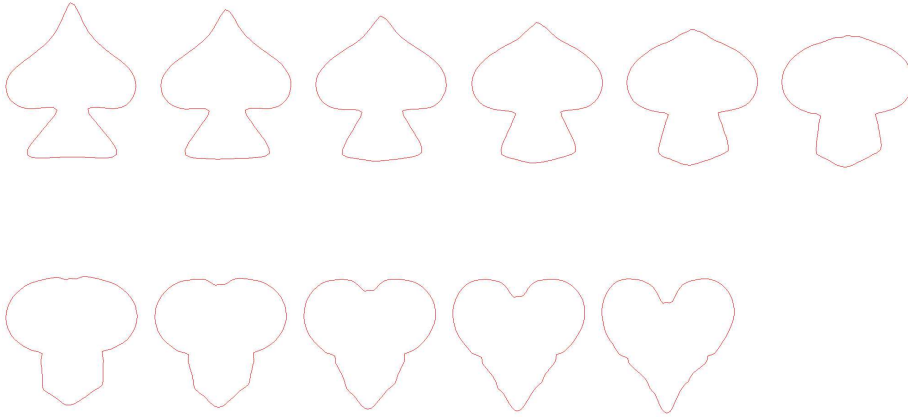


FIG. 7.5 – Evolution du contour

L'action d'un difféomorphisme g sur une matrice de covariance Σ centrée en C serait naturellement définie par $g.\Sigma = D_C g^{-1} \Sigma (D_C g^{-1})^T$. En reprenant le même raisonnement que dans 6.1, on obtiendrait une énergie liée au difféomorphisme du type :

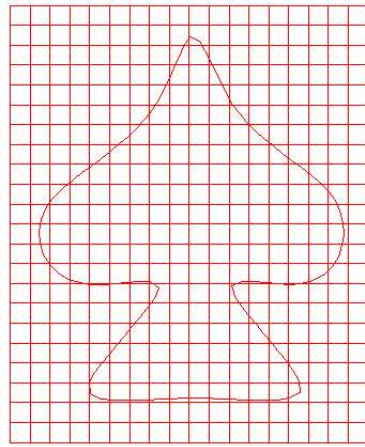
$$E_{diff}(A) = \int_0^1 \|v_t\|_Y^2 dt + \sum_n \int_0^1 \left\| \frac{dC_n}{dt} - v_t(C_n(t)) \right\|^2 dt \\ + \sum_n \int_0^1 \left\| \frac{d\Sigma_n}{dt} - D_{C_n(t)} v_t \Sigma_n - \Sigma_n (D_{C_n(t)} v_t)^T \right\|^2 dt$$

Il est possible de raffiner l'énergie en imposant aux matrices Σ_n d'être de forme allongée pour privilégier les quanta de segments ayant une orientation caractéristique.

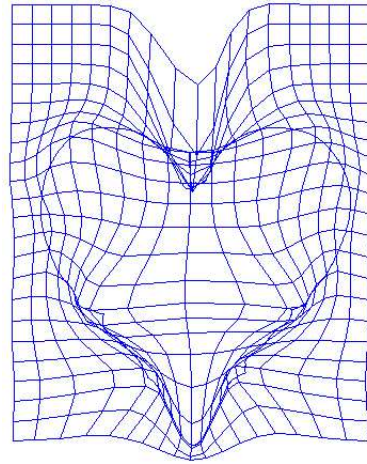
7.7 Conclusion

On a décrit dans ce chapitre une méthode originale de mise en correspondance de primitives. L'algorithme ne demande en plus que très peu de paramètres ou de connaissances *a priori* puisqu'il n'est notamment pas nécessaire de connaître les appariements de points. On remarque qu'il fournit des résultats de qualité comparable à ceux obtenus par mise en correspondance géodésique non ambiguë (i.e. avec donnée des correspondances entre points). Cette méthode est à comparer à des techniques de type "shape context matching" [7, 8] qui nécessite cependant un appariement des points définissant les formes à mettre en correspondance.

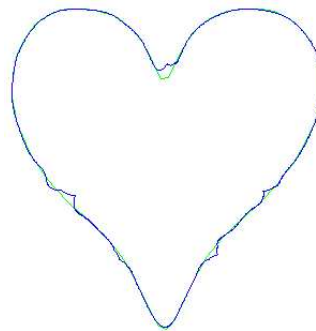
La méthode pourrait être étendue à d'autres formes de description de primitives (cf. 7.6). On voit qu'il serait également intéressant d'introduire une notion de multi-résolution (cf.4.3).



(a) Contour initial



(b) Contour final et déformation



(c) Adéquation du contour déformé (bleu) et du contour final (vert)

FIG. 7.6 – Résultat de la mise en correspondance par splines classiques

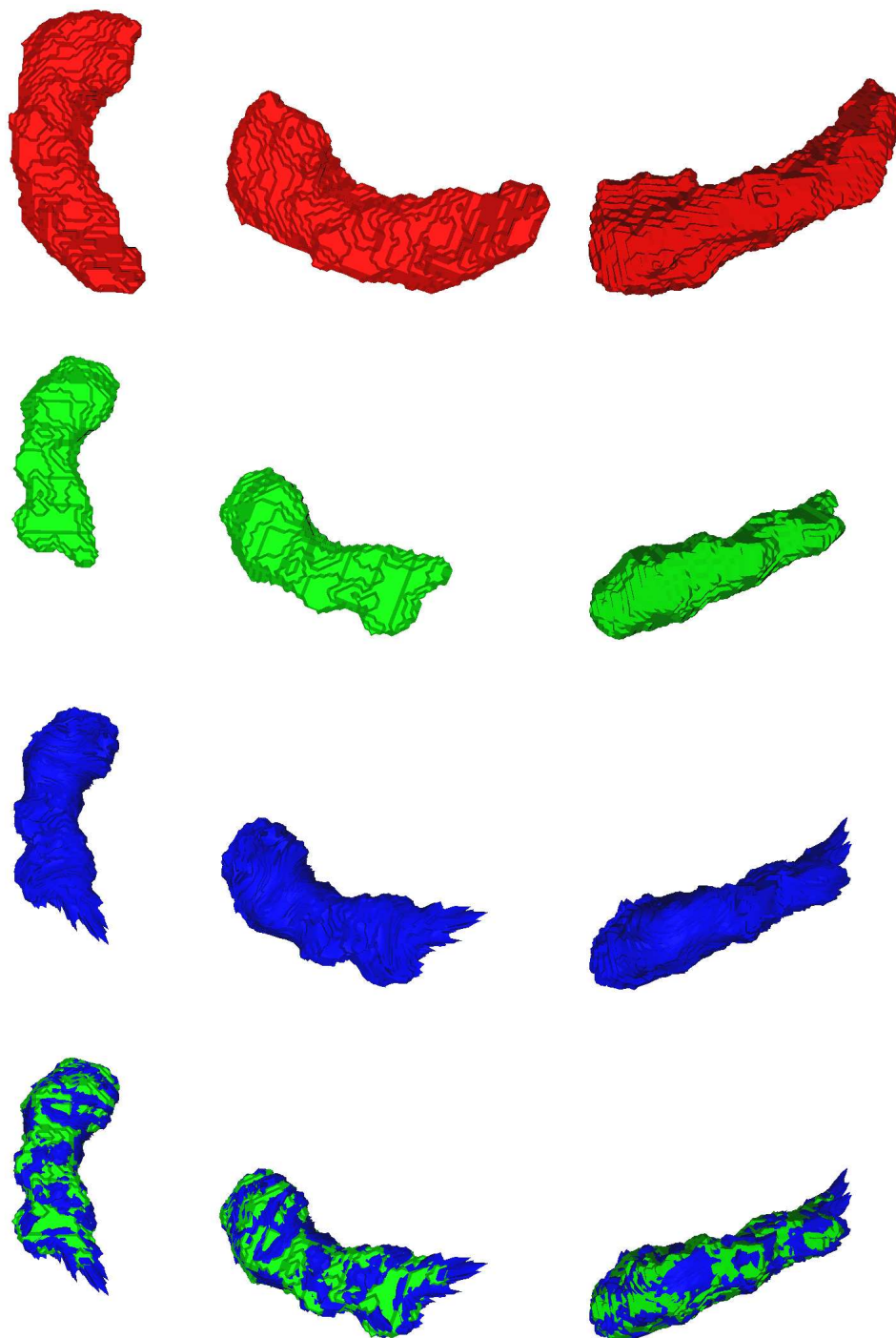


FIG. 7.7 – Résultat de la mise en correspondance d'hippocampes selon 3 points de vue (rouge : modèle, vert : cible, bleu : modèle déformé)

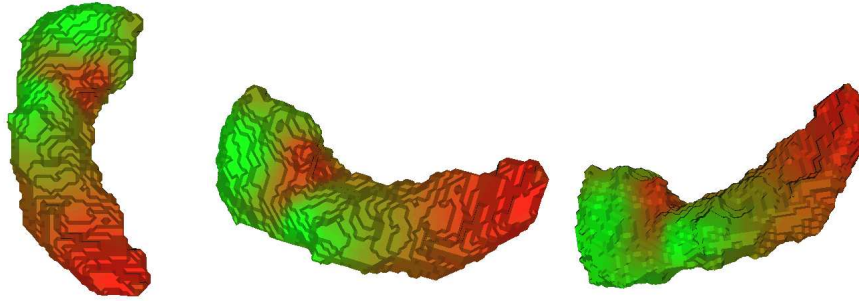
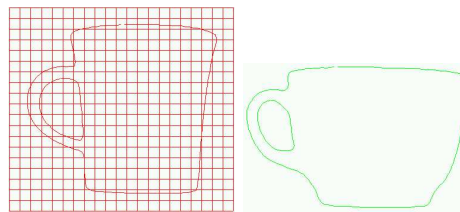
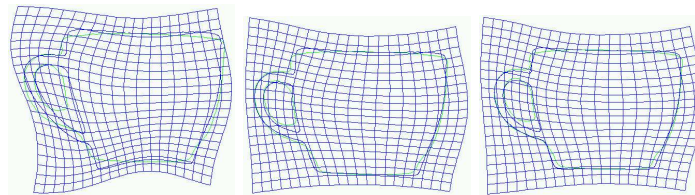


FIG. 7.8 – Déformation de la surface initiale (vert : faible déformation, rouge : forte déformation)



(a) Forme initiale

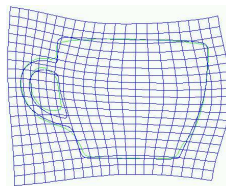
(b) Forme cible



(c) Déformation pour 10 quanta

(d) Déformation pour 20 quanta

(e) Déformation pour 50 quanta



(f) Déformation pour 100 quanta

FIG. 7.9 – Influence du nombre de quanta sur la mise en correspondance

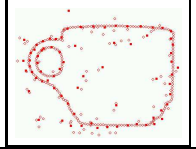
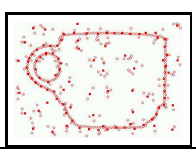
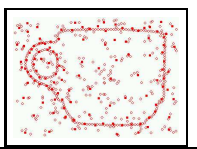
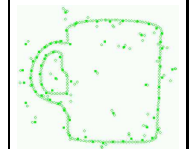
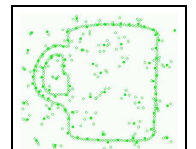
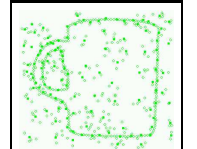
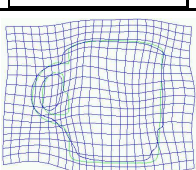
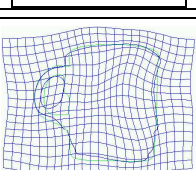
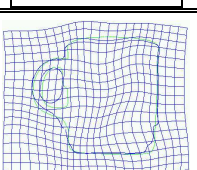
Nbre d'outliers	$N = 100$	$N = 200$	$N = 300$
Ensemble initial			
Ensemble cible			
Déformation			

FIG. 7.10 – Robustesse de l'algorithme en fonction du nombre d'outliers

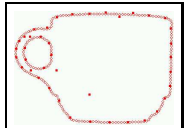
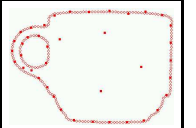
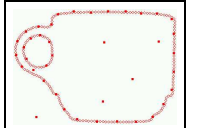



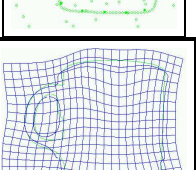
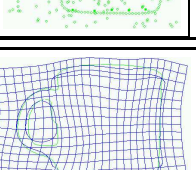
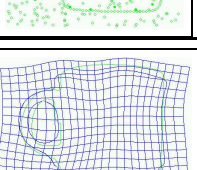
Nbre d'outliers	$N = 100$	$N = 300$	$N = 500$
Ensemble initial			
Ensemble cible			
Déformation			

FIG. 7.11 – Robustesse de l'algorithme en fonction du nombre d'outliers

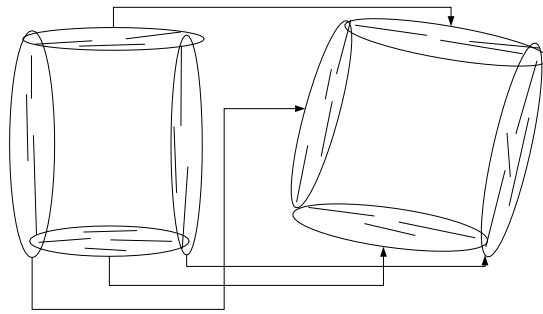


FIG. 7.12 – Les quanta à mettre en correspondance sont des ellipses

Chapitre 8

Conclusion

8.1 Mise en correspondance de surfaces

On a pu mettre en évidence l'apport de l'analyse des déformations dans une base adaptée ainsi que de la mise en correspondance multi-résolution au cadre théorique des grandes déformations. La définition de l'opérateur de régularisation nous a permis de contrôler de manière relativement précise la régularité de la déformation.

Il est intéressant de remarquer que notre algorithme est capable de retrouver des déformations relativement importantes avec précision sans aucune intervention d'un opérateur (donnée de *landmarks* par exemple) comme c'est le cas dans d'autres algorithmes. De plus, il comporte très peu de paramètres de réglage, ce qui le rend simple d'emploi.

Il serait aussi intéressant de tester ce type d'algorithme sur des données 3D avec des applications en imagerie médicale, par exemple. En effet, ce genre de données se prêterait très bien à cette modélisation de déformations souples.

8.2 Mise en correspondance de primitives

Dans le cadre de la mise en correspondance avec appariement (chap. 5), il faut noter qu'on n'a pas du tout tenu compte de l'aspect topologique et notamment des relations logiques entre les primitives (voisinage, intersection). La formulation de ces informations sous forme de graphe pourrait assez bien se prêter à la méthodologie employée dans notre travail [29, 56, 48, 47].

La mise en correspondance par quantification (chap. 7) est une approche intéressante mais il ne permet pour l'instant que d'apparier des objets définis à base de points bien qu'une extension à des objets plus complexes semble possible (cf. 7.6). On notera cependant que la mise en correspondance géodésique par quantification fournit d'aussi bons résultats que la mise en correspondance géodésique non ambigu (i.e. avec la donnée des correspondances entre les points). De plus, contrairement à la mise en correspondance avec appariement (chap. 5), il ne suppose pas une correspondance biunivoque entre les différentes primitives à apparier mais met en correspondance les ensembles dans leur globalité, permettant de faire abstraction des problèmes de fusion ou de division de primitives (cf. fig. 8.1), qu'on devrait, dans le cas de la mise en correspondance avec

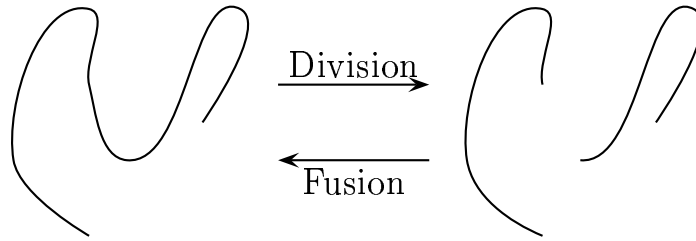


FIG. 8.1 – Fusion et division d'une primitive

appariement, gérer par des correspondances multiples. L'inconvénient est bien sûr qu'on ne profite justement pas de cette décomposition des objets à mettre en correspondance en des primitives isolées et que l'approche par quantification ne nous garantit pas la conservation de ces structures.

Une alternative à la mise en correspondance par quantification pourrait être la mise en correspondance de distributions, dont la philosophie est assez proche. Il s'agit là aussi non plus de mettre en correspondance des primitives une à une mais de les mettre en correspondance dans leur ensemble, l'ensemble des primitives étant modélisé par une distribution dont les singularités sont localisées sur les primitives.

Annexe A

Construction des MNE

On expliquera dans cette annexe comment sont construits les MNE à partir de plusieurs images aériennes. Pour cela, on utilise des méthodes de corrélation en se plaçant dans l'espace objet (3D) [45] plutôt qu'en coordonnées images. En effet, on évite alors le biais issu de la prise en compte d'une image comme image maîtresse et on évite la complexité qui proviendrait de la comparaison de toutes les images deux à deux.

A.1 Méthode de corrélation simple

On utilise une simple méthode de corrélation. On dispose de N clichés numériques ainsi que de leurs paramètres de prise de vue. On note p_1, \dots, p_N les transformations qui permettent de passer des coordonnées terrain aux coordonnées image (cf. fig. A.1).

$$z(x, y) = \underset{z \in [z_{min}; z_{max}]}{argmax} \left[\chi(I_1, \dots, I_N, x, y, z) \right]$$

$$\chi(I_1, \dots, I_N, x, y, z) = \iint_W \sum_{n=1}^N |I_n(p_n(x, y, z) + (u, v)) - \bar{I}(x, y, z, u, v)|^2 du dv$$

$$\bar{I}(x, y, z, u, v) = \frac{1}{N} \sum_{n=1}^N I_n(p_n(x, y, z) + (u, v))$$

$$E_C(z) = \sum_{i,j} \frac{\sum_{n=1}^P |I_n(p_n(x_{ij}, y_{ij}, z_{ij}) - \bar{I}(x_{ij}, y_{ij}, z_{ij}))|^2 v_n(x_{ij}, y_{ij}, z_{ij})}{\sum_{n=1}^P v_n(x_{ij}, y_{ij}, z_{ij})}$$

$$\bar{I}(x, y, z) = \frac{\sum_{n=1}^P I_n(p_n(x, y, z) v_n(x, y, z))}{\sum_{n=1}^P v_n(x, y, z)}$$

$$E_R(z) = \sum_{i,j} \Psi\left(z_{ij} - \frac{z_{i+1j} + z_{i-1j}}{2}\right)(1 - u_{ij}) + \Psi\left(z_{ij} - \frac{z_{ij+1} + z_{ij-1}}{2}\right)(1 - v_{ij})$$

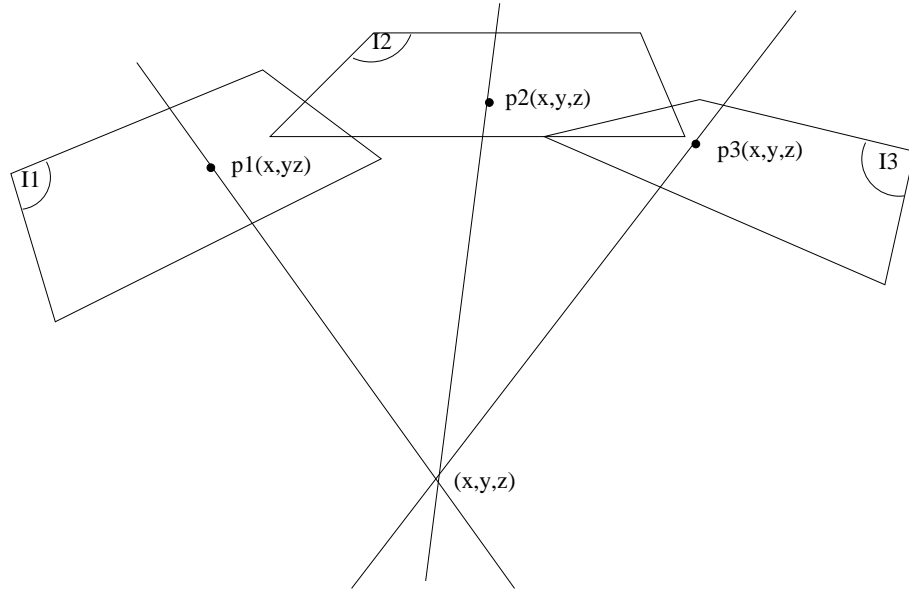


FIG. A.1 – Un point (x, y, z) et ses projections dans 3 images

A.2 Méthode par maxima locaux de corrélation

La méthode est à peu près la même que précédemment. La seule différence est que l'on stocke en chaque point (x, y) du MNE la série $(z_i(x, y))_{1 \leq i \leq N(x, y)}$ des maxima locaux de la corrélation en fonction de z . On effectue ensuite une relaxation probabiliste sur les $z_i(x, y)$ de manière à minimiser l'énergie :

$$E(S) = Corr(S) + \lambda Reg(S)$$

$$Corr(S) = \sum Corr(x, y, z(x, y))$$

$$Reg(S) = \sum \frac{1}{4} |4z(x, y) - (z(x-1, y) + z(x+1, y) + z(x, y-1) + z(x, y+1))|$$

La figure A.2 montre un exemple de MNE obtenu par ce type de méthode.



FIG. A.2 – MNE obtenu par maxima locaux de corrélation

Annexe B

Espaces de Hilbert à noyau reproduisant

On introduit dans cette annexe le cadre des espaces de Hilbert à noyau reproduisant [4, 1] dont on donnera d’abord une définition (B.1). On expliquera ensuite le lien entre un espace de Hilbert et son noyau reproduisant (B.2) et notamment la manière de construire un espace de Hilbert à noyau reproduisant à partir d’un certain type de noyau. On donnera ensuite une extension au premier ordre (B.3).

Cette théorie a été employée notamment dans le cadre des “support vector machines” [14, 50], des processus gaussiens [39, 58, 50] et de l’interpolation par splines [50, 57] (c’est ce cadre-là qui nous intéresse dans notre thèse).

On emploiera l’acronyme “RKHS” (Reproducing Kernel Hilbert Space) pour désigner un espace de Hilbert à noyau reproduisant.

B.1 De L^2 aux RKHS

On notera respectivement (\cdot, \cdot) et $|\cdot|$ le produit scalaire et la norme sur $L^2(\Omega)$, Ω étant un ouvert de \mathbb{R}^d . On se donne un espace fonctionnel de Hilbert \mathcal{H} de produit scalaire $\langle \cdot, \cdot \rangle$ et de norme $\|\cdot\|$. On veut trouver des éléments de \mathcal{H} qui jouent le même rôle pour le produit scalaire de \mathcal{H} que les masses de Dirac pour le produit scalaire usuel de L^2 .

C’est-à-dire que, de la même manière que, si δ_x est la masse de Dirac en un point $x \in \Omega$, on a pour tout $f \in L^2(\Omega)$

$$(\delta_x, f) = f(x)$$

on veut exhiber des éléments K_x de \mathcal{H} tels que, pour tout $h \in \mathcal{H}$

$$\langle K_x, h \rangle = h(x)$$

Ceci n’est possible que si \mathcal{H} est un RKHS. Donnons d’abord une définition générale des RKHS.

Définition B.1. *Un RKHS \mathcal{H} est un espace de Hilbert où les formes linéaires d’évaluation sont continues i.e. si et seulement si*

$$\forall x \in \Omega, \exists M_x < \infty, \forall f \in \mathcal{H}, |f(x)| < M_x \|f\|$$

Le théorème de Riesz nous dit alors qu'il existe pour chaque $x \in \Omega$, un élément K_x de \mathcal{H} tel que la forme linéaire $h \in \mathcal{H} \rightarrow h(x)$ soit représentée par cet élément K_x , i.e.

$$\forall h \in \mathcal{H}, h(x) = \langle K_x, h \rangle$$

Le fameux noyau reproduisant associé au RKHS \mathcal{H} est alors donné par la relation suivante :

$$K(x, y) = \langle K_x, K_y \rangle$$

Le théorème suivant [23] va nous servir à construire des RKHS à partir de noyaux vérifiant certaines conditions.

Théorème B.2. *Théorème de Mercer*

Si K est un noyau sur Ω symétrique, défini, positif, continu et vérifiant la condition suivante :

$$\int_{\Omega \times \Omega} K(x, y)^2 dx dy < \infty \quad (\text{B.1})$$

alors il existe une suite orthonormale de fonctions propres ϕ_i associées à des valeurs propres décroissantes λ_i telle que

$$K(x, y) = \sum_{i=0}^{\infty} \lambda_i \phi_i(x) \phi_i(y)$$

B.2 Cas usuel

B.2.1 Construction d'un RKHS à partir du théorème de Mercer

Définition B.3. Soit \mathcal{H} un espace de fonctions sur $\Omega \subset \mathbb{R}^d$ muni d'une structure hilbertienne. Un noyau K sur Ω est dit reproduisant pour \mathcal{H} si :

1. $\forall x \in \Omega, K_x = K(\cdot, x) \in \mathcal{H}$
2. $\forall x \in \Omega, \forall f \in \mathcal{H}, \langle f, K_x \rangle = f(x)$

Inversement, on peut définir un espace de Hilbert à noyau reproduisant à partir d'un noyau vérifiant certaines conditions.

Proposition B.4. Soit K un noyau symétrique, défini, positif, continu et vérifiant la condition (B.1). Alors il existe un espace de Hilbert $\mathcal{H}(K)$ à noyau reproduisant K .

Démonstration. Le théorème de Mercer (Thm. B.2) nous dit que le noyau K peut s'écrire sous la forme :

$$K(x, y) = \sum_{i=0}^{\infty} \lambda_i \phi_i(x) \phi_i(y)$$

où la suite (λ_i) est décroissante. On définit alors \mathcal{H}_k de la manière suivante :

$$\mathcal{H}(K) = \left\{ f \in L^2(\Omega) : \sum_{i=0}^{\infty} \frac{(f, \phi_i)^2}{\lambda_i} < \infty \right\} \quad (\text{B.2})$$

et on définit ainsi le produit scalaire :

$$\langle f, g \rangle = \sum_{i=0}^{\infty} \frac{(f, \phi_i)(g, \phi_i)}{\lambda_i} \quad (\text{B.3})$$

On a $(K_x, \phi_i) = \lambda_i \phi_i(x)$ de sorte que :

$$\sum_{i=0}^{\infty} \frac{(K_x, \phi_i)^2}{\lambda_i} = \sum_{i=0}^{\infty} \lambda_i \phi_i(x)^2 = K(x, x) < \infty$$

Ainsi les K_x appartiennent à $\mathcal{H}(K)$. De plus, on a pour $f \in \mathcal{H}(K)$:

$$\langle f, K_x \rangle = \sum_{i=0}^{\infty} \frac{(f, \phi_i)(K_x, \phi_i)}{\lambda_i} = \sum_{i=0}^{\infty} (f, \phi_i) \phi_i(x)$$

Comme on a l'inégalité suivante :

$$\left(\sum_{i=j}^k (f, \phi_i) \phi_i(x) \right)^2 \leq \sum_{i=j}^k \frac{(f, \phi_i)^2}{\lambda_i} \sum_{i=j}^k \lambda_i \phi_i(x)^2$$

et que les séries $\sum_{i=0}^{\infty} \frac{(f, \phi_i)^2}{\lambda_i}$ et $\sum_{i=0}^{\infty} \lambda_i \phi_i(x)^2$ convergent, la série $\sum_{i=0}^{\infty} \phi_i(x)(f, \phi_i)$ converge aussi de sorte que pour tout f dans \mathcal{H}_k , $\langle f, K_x \rangle = f(x)$. Le noyau K est donc bien reproduisant pour $\mathcal{H}(K)$. \square

On peut prouver que les éléments de $\mathcal{H}(K)$ sont continus. En effet, on peut écrire pour $f \in \mathcal{H}(K)$:

$$f(y) - f(x) = \sum_{i=0}^{\infty} (f, \phi_i) (\phi_i(y) - \phi_i(x))$$

et donc

$$\begin{aligned} (f(y) - f(x))^2 &\leq \sum_{i=0}^{\infty} \frac{(f, \phi_i)^2}{\lambda_i} \sum_{i=0}^{\infty} \lambda_i (\phi_i(y) - \phi_i(x))^2 \\ &\leq \|f\|^2 (K(y, y) - 2K(y, x) + K(x, x)) \end{aligned}$$

On conclut en invoquant le fait que K est continu.

B.2.2 Construction d'un RKHS à partir d'un noyau invariant par translation

Malheureusement, certains noyaux qu'on utilise dans notre travail ne vérifient pas la condition (B.1). Ceci est le cas quand $\Omega = \mathbb{R}^d$ et que K est un noyau invariant par translation. On peut tout de même construire un espace de Hilbert à noyau reproduisant K sous certaines conditions [59].

Proposition B.5. *Soit K un noyau invariant par translation sur \mathbb{R}^d i.e. $K(x, y) = k(x - y)$. S'il existe une fonction F paire, intégrable et positive telle que :*

$$k(u) = \int_{\mathbb{R}^d} e^{-iu^T v} F(v) dv$$

alors il existe un espace de Hilbert $\mathcal{H}(K)$ à noyau reproduisant K .

Démonstration. On définit la fonction \tilde{F} telle que $\tilde{F}(v) = \sqrt{F(v)}$ et on pose :

$$\tilde{k}(u) = \int_{\mathbb{R}^d} e^{-iu^T v} \tilde{F}(v) dv$$

On définit alors le noyau \tilde{K} tel que $\tilde{K}(x, y) = \tilde{k}(x - y)$ (en quelque sorte, la “racine carrée” du noyau K). On définit alors ainsi l'espace $\mathcal{H}(K)$:

$$\mathcal{H}(K) = \left\{ f \in L^2(\mathbb{R}^d) : \exists g \in L^2(\mathbb{R}^d), f = \tilde{K}g \right\}$$

Prouvons tout d'abord que l'application de $L^2(\mathbb{R}^d)$ dans $\mathcal{H}(K)$ qui à f associe $\tilde{K}f$ est un isomorphisme. Soit donc $f \in L^2(\mathbb{R}^d)$ telle que $\tilde{K}f = 0$. Ceci signifie que $\tilde{k} * f = 0$. On notera, \hat{h} la transformée de Fourier d'un élément h de $L^2(\mathbb{R}^d)$ et \check{h} sa transformée inverse. On a donc $\tilde{F}\check{f} = 0$. Comme \tilde{F} ne s'annule pas, \check{f} est identiquement nulle et $f = 0$. On peut alors définir une structure hilbertienne sur $\mathcal{H}(K)$ en posant :

$$\langle f, f' \rangle = (g, g'), f = \tilde{K}g, f' = \tilde{K}g'$$

De plus, on a $k = \hat{F}$ et $\tilde{k} = \hat{\tilde{F}}$ et $F = \tilde{F}^2$. Donc $k = \tilde{k} * \tilde{k}$. Si on note $K_x = K(\cdot, x)$ et $\tilde{K}_x = \tilde{K}(\cdot, x)$, ceci signifie que $K_x = \tilde{K}\tilde{K}_x$ et K_x appartient bien à $\mathcal{H}(K)$.

Enfin, si $f \in \mathcal{H}(K)$, alors il existe $g \in L^2(\mathbb{R}^d)$ tel que $f = \tilde{K}g$. Comme on sait d'autre part que $K_x = \tilde{K}\tilde{K}_x$, on peut écrire :

$$\langle f, K_x \rangle = (g, \tilde{K}_x) = (\tilde{K}g)(x) = f(x)$$

On déduit de toutes les remarques précédentes que K est bien un noyau reproduisant pour $\mathcal{H}(K)$. \square

B.3 Extension au premier ordre

B.3.1 Construction d'un RKHS de premier ordre à partir du théorème de Mercer

Définition B.6. Soit \mathcal{H} un espace de fonctions sur Ω muni d'une structure hilbertienne. Un noyau K sur Ω est dit reproduisant de premier ordre pour \mathcal{H} si :

1. $\forall x \in \Omega, K_x = K(\cdot, x) \in \mathcal{H}$
2. $\forall x \in \Omega, \forall f \in \mathcal{H}, \langle f, K_x \rangle = f(x)$
3. $\forall y \in \Omega, \forall u \in \mathbb{R}^d, K'_{y,u} = \partial_2 K(\cdot, y) \cdot u = \partial_1 K(y, \cdot) \cdot u \in \mathcal{H}$
4. $\forall y \in \Omega, \forall u \in \mathbb{R}^d, \forall f \in \mathcal{H}, \langle f, K'_{y,u} \rangle = d_y f \cdot u$

On peut à nouveau construire un espace de Hilbert à partir d'un noyau vérifiant certaines conditions comme dans Prop. B.4.

Proposition B.7. Soit K un noyau sur Ω de classe C^2 , symétrique, défini, positif et vérifiant la condition (B.1). Alors il existe un espace de Hilbert $\mathcal{H}^1(K)$ à noyau reproduisant de premier ordre K .

Démonstration. On peut à nouveau appliquer le théorème de Mercer (Thm. B.2) :

$$K(x, y) = \sum_{i=0}^{\infty} \lambda_i \phi_i(x) \phi_i(y)$$

On définit $\mathcal{H}^1(K)$ et son produit scalaire de la même manière que précédemment (cf. (B.2) et (B.3)). Les K_x appartiennent donc bien encore à $\mathcal{H}^1(K)$. Comme K est C^2 et que $\phi_i = \frac{K\phi_i}{\lambda_i}$, les ϕ_i sont aussi C^2 . On peut alors écrire :

$$(K'_{x,u}, \phi_i) = \int_{\Omega} \partial_1 K(x, y) \cdot u \phi_i(y) dy = \frac{d}{dx} \left(\int_{\Omega} K(x, y) \phi_i(y) dy \right) \cdot u = \lambda_i d_x \phi \cdot u$$

de sorte que :

$$\sum_{i=0}^{\infty} \frac{(K'_{x,u}, \phi_i)^2}{\lambda_i} = \sum_{i=0}^{\infty} \lambda_i (d_x \phi \cdot u)^2 = \partial_1 \partial_2 K(x, x)(u, u) < \infty$$

Les $K'_{x,u}$ appartiennent donc bien à $\mathcal{H}^1(K)$. Comme précédemment, on a $\langle f, K_x \rangle = f(x)$ pour $f \in \mathcal{H}^1(K)$. Et on peut aussi écrire :

$$\langle f, K'_{x,u} \rangle = \sum_{i=0}^{\infty} \frac{(f, \phi_i)(K'_{x,u}, \phi_i)}{\lambda_i} = \sum_{i=0}^{\infty} (f, \phi_i) d_x \phi_i \cdot u$$

Comme on a l'inégalité suivante :

$$\left(\sum_{i=j}^k (f, \phi_i) d_x \phi_i \cdot u \right)^2 \leq \sum_{i=j}^k \frac{(f, \phi_i)^2}{\lambda_i} \sum_{i=j}^k \lambda_i (d_x \phi_i \cdot u)^2$$

la série $\sum_{i=0}^{\infty} (f, \phi_i) d_x \phi_i \cdot u$ converge de sorte que pour tout f dans \mathcal{H}_k , $\langle f, K'_{x,u} \rangle = d_x f \cdot u$. Le noyau K est donc bien reproduisant de premier ordre pour $\mathcal{H}^1(K)$. \square

On peut prouver que les éléments de $\mathcal{H}(K)$ sont C^1 . Montrons d'abord qu'ils sont différentiables. On peut écrire pour $f \in \mathcal{H}^1(K)$:

$$\frac{f(x+u) - f(x) - \sum_{i=0}^{\infty} (f, \phi_i) d_x \phi_i \cdot u}{|u|} = \sum_{i=0}^{\infty} (f, \phi_i) \frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|}$$

On va d'abord montrer que la série du terme de droite converge. En effet, on a l'inégalité suivante :

$$\left(\sum_{i=j}^k (f, \phi_i) \frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|} \right)^2 \leq \sum_{i=j}^k \frac{(f, \phi_i)^2}{\lambda_i} \sum_{i=j}^k \lambda_i \left(\frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|} \right)^2$$

Or la série $\sum_{i=0}^{\infty} \frac{(f, \phi_i)^2}{\lambda_i}$ converge et le développement du carré dans :

$$\sum_{i=0}^{\infty} \lambda_i \left(\frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|} \right)^2$$

nous montre que cette série peut s'écrire sous la forme d'une somme de séries convergentes et converge donc aussi, ce qui nous prouve que c'est également le cas pour la série $\sum_{i=0}^{\infty} (f, \phi_i) \frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|}$. On est alors en droit d'écrire :

$$\begin{aligned} \left(\sum_{i=0}^{\infty} (f, \phi_i) \frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|} \right)^2 &\leq \sum_{i=0}^{\infty} \frac{(f, \phi_i)^2}{\lambda_i} \sum_{i=0}^{\infty} \lambda_i \left(\frac{\phi_i(x+u) - \phi_i(x) - d_x \phi_i \cdot u}{|u|} \right)^2 \\ &= \|f\|^2 \frac{\epsilon(x, u)}{|u|^2} \end{aligned}$$

avec :

$$\begin{aligned} \epsilon(x, u) &= K(x+u, x+u) + K(x, x) + \partial_1 \partial_2 K(x, x)(u, u) \\ &\quad - 2K(x, x+u) + 2\partial_1 K(x, x) \cdot u - 2\partial_1 K(x, x+u) \cdot u \end{aligned}$$

Un rapide calcul utilisant le fait que K est C^2 prouve que $\epsilon(x, u) = o(|u|^2)$. On en déduit finalement que f est différentiable et que

$$d_x f = \sum_{i=0}^{\infty} (f, \phi_i) d_x \phi_i$$

On peut maintenant écrire :

$$d_y f \cdot v - d_x f \cdot v = \sum_{i=0}^{\infty} (f, \phi_i) (d_y \phi_i \cdot v - d_x \phi_i \cdot v)$$

ce qui entraîne :

$$(d_y f \cdot v - d_x f \cdot v)^2 \leq \|f\|^2 (\partial_1 \partial_2 K(y, y)(v, v) - 2\partial_1 \partial_2 K(x, y)(v, v) + \partial_1 \partial_2 K(x, x)(v, v))$$

On conclut que f est C^1 en invoquant le fait que K est C^2 et que le terme de droite tend donc vers 0 quand v tend vers 0.

B.3.2 Construction d'un RKHS de premier ordre à partir d'un noyau invariant par translation

On se place dans les mêmes conditions que dans B.2.2. On peut encore énoncer une proposition du même type que B.5.

Proposition B.8. *Soit K un noyau invariant par translation sur \mathbb{R}^d i.e. $K(x, y) = k(x - y)$. S'il existe une fonction F paire, intégrable et positive telle que :*

$$k(u) = \int_{\mathbb{R}^d} e^{-iu^T v} F(v) dv$$

et telle que $v \rightarrow v\sqrt{F(v)}$ soit aussi intégrable, alors il existe un espace de Hilbert $\mathcal{H}(K)$ à noyau reproduisant K .

Démonstration. On reprend les mêmes notations que dans la démonstration de B.5 et on définit $\mathcal{H}^1(K)$ de la même manière que $\mathcal{H}(K)$. Le raisonnement précédent peut encore s'appliquer. Les K_x appartiennent donc à $\mathcal{H}^1(K)$ et $\langle f, K_x \rangle = f(x)$ pour tout $f \in \mathcal{H}^1(K)$.

On sait par ailleurs que \tilde{k} est différentiable puisque $v \rightarrow vG(v)$ est intégrable, donc on peut écrire :

$$K'_{y,u}(x) \int_{\mathbb{R}^d} \tilde{K}(x,z) \partial_2 \tilde{K}(z,y) \cdot u dy = \tilde{K} \tilde{K}'_{y,u}$$

Ceci prouve que les $K'_{x,u}$ appartiennent à $\mathcal{H}^1(K)$.

De plus, si $f = \tilde{K}g$ appartient à $\mathcal{H}^1(K)$, on peut écrire :

$$\langle K'_{x,u}, f \rangle = \langle \tilde{K}'_{y,u}, f \rangle = \frac{d}{dx} (\tilde{K}_x, g) \cdot u = d_x f \cdot u$$

On en déduit que le noyau K est reproduisant de premier ordre pour $\mathcal{H}^1(K)$. \square

Table des figures

2.1	Action d'une rotation sur un 7-uplet de \mathbb{R}^2	12
2.2	Chemin \mathbf{J} d'images et chemin \mathbf{g} des difféomorphismes associés . .	20
2.3	Trajectoire d'un point	24
3.1	Deux possibilités : modifier l'altimétrie ou la planimétrie	28
3.2	Influence du poids de régularisation λ sur la mise en correspon- dance du petit disque et du grand disque	29
3.3	Morphing avec un opérateur de régularisation gaussien. Les images entre la première et la dernière images sont calculées par l'algo- rithme.	30
3.4	Influence de la taille du noyau gaussien régularisant sur la mise en correspondance du petit disque et du grand disque (taille de l'image 64 pixels)	31
3.5	Comparaison des différents types de mise en correspondance. Les images entre la première et la dernière de chaque série sont cal- culées par l'algorithme	42
3.6	Comparaison des déformations pour les différents types de mise en correspondance	43
3.7	Comparaison des déformations pour les différents types de mise en correspondance (suite)	44
3.8	Dans chaque paire : image finale transformée par la déformation inverse (à comparer à l'image initiale) et image initiale transfor- mée par la déformation directe (à comparer à l'image finale) . . .	45
3.9	Variation des énergies au cours de l'algorithme à base d'ondelettes avec multi-résolution (continu) et sans multi-résolution (pointillés)	46
3.10	Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opé- rateur de régularisation $\beta = 0$	47
3.11	Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opé- rateur de régularisation $\beta = 0.5$	48
3.12	Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opé- rateur de régularisation $\beta = 1$	49
3.13	Résultat de l'algorithme multi-résolution à différentes échelles. Facteur de croissance exponentielle des valeurs propres de l'opé- rateur de régularisation $\beta = 1.5$	50
3.14	Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 0$)	51

3.15	Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 0.5$)	51
3.16	Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 1$)	52
3.17	Jacobiens des déformations à différentes échelles de l'algorithme multi-résolution ($\beta = 1.5$)	52
3.18	Comparaison de l'image initiale et de l'image initiale déformée sans apprentissage (fig. 3.18(c)) et avec apprentissage (fig. 3.18(d)).	53
4.1	Une courbe définie par des points et des tangentes	55
5.1	Mise en correspondance hiérarchique de deux ensembles	62
5.2	Hierarchie des matrices d'appariement	63
5.3	Distance du point P_a à la droite $F^n Q_b^n$	70
5.4	Appariement de points en multi-vues : image 1. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.	71
5.5	Appariement de points en multi-vues : image 2. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.	72
5.6	Appariement de points en multi-vues : image 3. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.	72
5.7	Appariement de points en multi-vues : image 4. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.	73
5.8	Appariement de points en multi-vues : image 5. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.	73
5.9	Appariement de points en multi-vues : image 6. Les points des figures 5.4 à 5.9 portant le même numéro sont mis en correspondance.	74
5.10	Segment S_a et ses tangentes aux points interpolants ($N_{int} = 2$) .	78
5.11	Deux ensembles de segments avec $N = 20$, $n_1 = n_2 = 20$	80
5.12	Appariement des segments trouvée par les deux algorithmes. Plus le pixel est foncé, plus la probabilité d'appariement est élevée. La dernière ligne et la dernière colonne représentent les outliers. . .	80
5.13	Mise en correspondance du modèle par les deux algorithmes . . .	81
5.14	Retrouver un bâtiment à partir d'un modèle	82
5.15	Première image stéréoscopique. Ensemble de segments de départ	83
5.16	Seconde image stéréoscopique. Ensemble de segments d'arrivée. .	84
5.17	Mise en correspondance de lignes	85
5.18	Mise en correspondance de polygones	86
5.19	Une facette définie par ses sommets et son vecteur normal	86
5.20	Retrouver le bon objet 3D	87
6.1	Mise en correspondance de contours	101
6.2	Evolution du contour au cours du temps	102
6.3	Mise en correspondance de contours	103
6.4	Evolution du contour au cours du temps	104

7.1	Les deux ensembles à mettre en correspondance, les centres de quanta et leurs trajectoires	108
7.2	Passage continu d'une géodésique à une non-géodésique	111
7.3	Mise en correspondance de mots manuscrits	113
7.4	Résultat de la mise en correspondance par splines géodésiques	114
7.5	Evolution du contour	115
7.6	Résultat de la mise en correspondance par splines classiques	116
7.7	Résultat de la mise en correspondance d'hippocampes selon 3 points de vue (rouge : modèle, vert : cible, bleu : modèle déformé)	117
7.8	Déformation de la surface initiale (vert : faible déformation, rouge : forte déformation)	118
7.9	Influence du nombre de quanta sur la mise en correspondance	118
7.10	Robustesse de l'algorithme en fonction du nombre d'outliers	119
7.11	Robustesse de l'algorithme en fonction du nombre d'outliers	119
7.12	Les quanta à mettre en correspondance sont des ellipses	120
8.1	Fusion et division d'une primitive	122
A.1	Un point (x, y, z) et ses projections dans 3 images	124
A.2	MNE obtenu par maxima locaux de corrélation	125

List of Algorithms

1	Mise en correspondance de deux images I_0 et I_1 en multi-résolution	35
2	Algorithme de Sinkhorn pour une matrice M de taille $(K+1, L+1)$ (outliers : dernière ligne et dernière colonne)	66
3	Mise à jour en cascade des correspondances	67
4	Mise en correspondance hiérarchique	67
5	Mise en correspondance de points en multivues	70
6	Mise en correspondance d'objets définis par des points et des vecteurs	78
7	Mise en correspondance géodésique	98
8	Algorithme de Soft K-means	107
9	Quantification et mise en correspondance géodésique	112

Bibliographie

- [1] N. Akhiezer and I. Glazman. *Theory of Linear Operators in Hilbert Space*. Dover Publications, 1993.
- [2] L. Alvarez, J. Weickert, and J. Sanchez. A scale-space approach to nonlocal optical flow calculations. In *Scale-Space Theories in Computer Vision*, pages 235–246, 1999.
- [3] Y. Amit. A nonlinear variational problem for image matching. *SIAM Journal on Scientific Computing*, 15(1) :207–224, 1994.
- [4] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, pages 337–404, 1950.
- [5] R. Bajcsy and S. Kovačič. Multiresolution elastic matching. *Computer Vision and Graphical Image Processing*, 46(1) :1–21, 1989.
- [6] J. Barron, D. Fleet, S. Beauchemin, and T. Burkitt. Performance of optical flow techniques. *CVPR*, 92 :236–242, 1994.
- [7] S. Belongie, J. Malik, and J. Puzicha. Matching shapes. In *ICCV01*, pages 454–463, July 2001.
- [8] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts, 2001.
- [9] C. Bernard. Fast optic flow computation with discrete wavelets. Technical Report RI365, Centre de Mathématiques Appliquées, École Polytechnique, May 1997.
- [10] C. P. Bernard. Discrete wavelet analysis for fast optic flow computation. *Applied and Computational Harmonic Analysis*, 11(1) :32–63, July 2001.
- [11] L. Bookstein. *Morphometric tools for landmark data, geometry and biology*. Cambridge University Press, 1991.
- [12] J. M. Buhmann. *The Handbook of Brain Theory and Neural Networks*. MIT Press, 2002. Data Clustering and Learning.
- [13] J. M. Buhmann and T. Hoffman. A maximum entropy approach to pairwise data clustering. In *Proceedings of the International Conference on Pattern Recognition*, volume II, pages 207–212. ICPR, IEEE Computer Society Press, 1994.
- [14] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2) :121–167, 1998.
- [15] G. Chiarpato, O. Faugeras, and R. Keriven. Approximations of shape metrics and application to shape warping and empirical shape statistics. Technical Report 4820, INRIA, INRIA, 2004 route des Lucioles, BP 93, 06902 Sophia Antipolis, France, May 2003.

- [16] G. E. Christensen, P. Yin, M. W. Vannier, K. S. C. Chao, J. F. Dempsey, and J. F. Williamson. Large-deformation image registration using fluid landmarks. In *Proceedings of the 4th IEEE Southwest Symposium on Image Analysis and Interpretation*, page 269. IEEE Computer Society, 2000.
- [17] H. Chui and A. Rangarajan. A new algorithm for non-rigid point matching. In *Proceedings of CVPR*, volume 2, pages 40–51. Conference on Computer Vision and Pattern Recognition, June 2000.
- [18] H. Chui and A. Rangarajan. Learning an atlas from unlabeled point-sets. In *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pages 58–65. IEEE, IEEE Press, 2001.
- [19] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Conference on Computer Vision and Image Understanding*, 89(2/3) :114–141, 2003.
- [20] H. Chui, L. Win, R. Schultz, J. Duncan, and A. Rangarajan. A unified non-rigid feature registration method for brain mapping. *Medical Image Analysis*, 2002.
- [21] H. Chui, L. Win, R. Schultz, J. Duncan, and A. Rangarajan. Unsupervised learning of an atlas from unlabeled point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004.
- [22] R. T. Collins. A space-sweep approach to true multi-image matching. In *Proceedings of Computer Vision and Pattern Recognition Conference*, pages 358–363. CVPR, 1996.
- [23] R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume I. John Wiley and Sons, 1989.
- [24] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39 :1–38, 1977. Series B.
- [25] P. Dupuis and U. Grenander. Variational problems on flows of diffeomorphisms for image matching, 1998.
- [26] E. W. Forgy. Cluster analysis of multivariate data : Efficiency versus interpretability of classifications. In *Biometric Society Meetings*, volume 21, page 768, 1965.
- [27] L. Garcin, A. Rangarajan, and L. Younes. Non rigid registration of shapes via diffeomorphic point matching and clustering. Preprint, 2004.
- [28] S. Gold, C. P. Lu, A. Rangarajan, S. Pappu, and E. Mjolsness. New algorithms for 2D and 3D point matching : Pose estimation and correspondence. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, pages 957–964. The MIT Press, 1995.
- [29] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4) :377–388, 1996.
- [30] H. Greenspan, J. Goldberger, and L. Ridel. A continuous probabilistic framework for image matching. *Computer Vision and Image Understanding*, 84(3) :384–406, 2001.
- [31] U. Grenander. *General Pattern Theory*. Oxford University Press, 1993.

- [32] G. Hermosillo, C. Chefd'Hotel, and O. Faugeras. A variational approach to multi-modal image matching. Technical Report 4117, INRIA, 2001.
- [33] B. Horn and B. Schunck. Determining optical flow. *AI*, 17(1-3) :185–203, August 1981.
- [34] S. Jaffard and P. Laurençot. *Orthonormal wavelets, analysis of operators, and applications to numerical analysis*, pages 543–601. Academic Press Professional, Inc., 1992.
- [35] S. Joshi and M. Miller. Landmark matching via large deformation diffeomorphisms. *IEEE transactions in image processing*, 9(8) :1357–1370, 2000.
- [36] S. C. Joshi and M. I. Miller. Landmark matching via large deformation diffeomorphisms. *IEEE Transactions On Image Processing*, 9(8) :1357–1370, August 2000.
- [37] M. Lefébure and L. D. Cohen. Image registration, optical flow and local rigidity. *Journal of Mathematical Imaging and Vision*, 14(2) :131–147, 2001.
- [38] S. P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28 :129–137, 1982.
- [39] D. J. C. MacKay. Introduction to Gaussian processes. In C. M. Bishop, editor, *Neural Networks and Machine Learning*, NATO ASI Series, pages 133–166. Kluwer Academic Press, 1998.
- [40] S. G. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2) :617–643, 1992.
- [41] Y. Meyer. Wavelets and operators. In *Analysis at Urbana, Vol. I (Urbana, IL, 1986–1987)*, volume 137 of *London Mathematical Society Lecture Note Series*, pages 256–365. Cambridge University Press, Cambridge, 1989.
- [42] I. Miller, M and L. Younes. Group action, diffeomorphisms and matching : a general framework. *International Journal of Computer Vision*, 41 :61–84, 2001. (Originally published in electronic form in : *Proceeding of SCTV 99*, <http://www.cis.ohio-state.edu/szhu/SCTV99.html>).
- [43] M. I. Miller, S. C. Joshi, and G. E. Christensen. Large deformation fluid diffeomorphisms for landmark and image matching. In A. W. Toga, editor, *Brain Warping*, chapter 7. Academic Press, 1999.
- [44] R. Neal and G. Hinton. *Learning in Graphical Models*, chapter A view of the EM algorithm that justifies incremental, sparse, and other variants, pages 355–368. MIT Press, Cambridge, MA, 1999.
- [45] N. Paparoditis, C. Thom, and H. Jibrini. Surface reconstruction of urban area from highly overlapping digital aerial images. In *International Archives of Photogrammetry and Remote Sensing*, volume XXXIII. ISPRS, July 2000.
- [46] A. Rangarajan, H. Chui, and E. Mjolsness. A new distance measure for non-rigid image matching. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 237–252. Conference on Computer Vision and Pattern Recognition, 1999.
- [47] A. Rangarajan and E. Mjolsness. A lagrangian relaxation network for graph matching. *IEEE Transactions on Neural Networks*, 7(6) :1365–1381, 1996.
- [48] A. Rangarajan, A. L. Yuille, and E. Mjolsness. Convergence properties of the softassign quadratic assignment algorithm. *Neural Computation*, 11(6) :1455–1474, 1999.

- [49] C. Schmid and A. Zisserman. Automatic line matching across views. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 666. IEEE Computer Society, 1997.
- [50] M. Seeger. Relationships between gaussian processes, support vector machines and smoothing splines.
- [51] E. P. Simoncelli. Bayesian multi-scale differential optical flow. In B. Jähne, H. Haussecker, and P. Geissler, editors, *Handbook of Computer Vision and Applications*, volume 2, pages 397–422. Academic Press, San Diego, 1999.
- [52] R. Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Annals of Mathematical Statistics*, 35 :876–879, 1964.
- [53] A. Trouvé. An approach of pattern recognition through infinite dimensional group. Technical Report 9, Département de Mathématiques et Informatiques, ENS Ulm, Paris, 1995.
- [54] A. Trouvé. Diffeomorphism groups and pattern matching in image analysis. *International Journal of Computer Vision*, 28(3) :213–221, 1998.
- [55] A. Trouvé and L. Younes. Metamorphoses through lie group action. *Foundations of Computational Mathematics*, 2004.
- [56] M. A. van Wyk, T. S. Durrani, and B. J. van Wyk. A RKHS interpolator-based graph matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7) :988–995, 2002.
- [57] G. Wahba. *Spline Models for Observational Data*, volume 59. Society for Industrial and Applied Mathematics, Philadelphia, 1990.
- [58] C. K. I. Williams and C. E. Rasmussen. Gaussian processes for regression. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Proceedings of Conference on Advances in Neural Information Processing Systems, NIPS*, volume 8. MIT Press, 1995.
- [59] L. Younes. Deformation analysis for shape and image processing. Notes de cours.
- [60] L. Younes. Computable elastic distances between shapes. *SIAM Journal on Applied Mathematics*, 58(2) :565–586, 1998.
- [61] L. Younes and V. Camion. Geodesic interpolating splines. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 513–527. CVPR, Springer, 2001.
- [62] L. Younes and A. Trouvé. Local geometry of deformable templates. Preprint.
- [63] Z. Zhang. Estimating motion and structure from correspondences of line segments between two perspective images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12) :1129–1139, 1995.