Mesh deformation strategies in shape optimization. Application to forensic facial reconstruction

Chiara Nardoni

Université Pierre et Marie Curie Paris 6

Institut des Sciences du Calcul et des Données Fed 3

Laboratoire Jacques-Louis Lions UMR 7598

Thèse de Doctorat de
l'Université Pierre et Marie Curie

pour l'obtention du grade de

Docteur de l'Université Pierre et Marie Curie

Spécialité : Mathématiques Appliquées

par

Chiara NARDONI

Mesh deformation strategies in shape optimization

Application to forensic facial reconstruction

après avis des rapporteurs

M. Bruno Lévy & M. Gabriel Peyré

devant le jury composé de

Mme.	Maya de Buhan	Directeur de Thèse
Mme.	Marion Darbas	Examinateur
M.	Pascal Frey	Directeur de Thèse
M.	Bruno Lévy	Rapporteur
M.	Yvon Maday	Examinateur
M.	Gabriel Peyré	Rapporteur
M.	Stefan Schlager	Examinateur

École Doctorale de Sciences Mathématiques de Paris Centre ED 386

Faculté de Mathématiques UFR 929

Chiara NARDONI :

Sorbonne Universités, UPMC Univ Paris 06, UMR 7598, Laboratoire Jacques-Louis Lions, F-75005, Paris, France.
Sorbonne Universités, UPMC Univ Paris 06, UMR 7598, ISCD, Paris, France.

Adresse électronique: chiara.nardoni@upmc.fr.

Remerciements

Cette thèse de doctorat a constitué le fil conducteur d'une tranche de vie de son auteur. De nombreuses personnes se sont retrouvés ainsi dans mon chemin; ce sont certaines de ces personnes que j'aimerais mettre en avant dans ces remerciements.

Je remercie en premier lieu Pascal de m'avoir permis des travailler dans d'aussi bonnes conditions, de son intérêt et son soutien, de ses nombreux et sages conseils durant la rédaction de ma thèse.

Je remercie Maya d'avoir accepté la tâche d'encadrer cette thèse. Ton soutien, le nombre d'heures que tu as dédié a travailler avec moi, ta finesse scientifique m'ont beaucoup apporté. Merci de m'avoir fait découvrir le metier de chercheuse et de m'avoir transmis toute ta passion.

Je suis très honorée que Bruno Lévy et Gabriel Peyré dont le travail scientifique j'apprécie énormément aient accepté de consacrer leur temps à le lecture de cette thèse. Je remercie vivement Yvon Maday, Marion Darbas et Stefan Schlager d'avoir accepté de faire partie de mon jury.

Ce travail n'aurait pas été possible sans le soutien de l'Université Sorbonne et de la Chaire Thématique Facile, qui m'ont permis, grâce à une allocation de recherches et diverses aides financières, de me consacrer sereinement à l'élaboration de mes travaux. Je remercie Charles, toujours à l'écoute de mes questions, de m'avoir transmis un peu de son savoir.

Je remercie également les partenaires du projet FaciLe pour les nombreuses échanges et reunions scientifiques et de m'avoir fait découvrir une application interessante des mathématiques. Je remercie tous les collègues que dans ces trois années ont fait parti de mon quotidien, Silvia, Chantal, Jérémy F., Robin, Jérémy D., Titien, Lucille, Luis, Nicolas, Paule Krystelle, Delphine, Sophie, Halimatou pour que ce soit d'une séance

de vapotage, d'un atelier d'impro ou d'un calcul de dérivée de forme autour de la défunte machine à m&m's. Merci à Sarah, Thomas, Auréliane, Loic, Jérémy pour les nombreux conseils sur la préparation de cette soutenance. Je remercie vivement Agnieszka pour son travail de valorisation, son soutien et ses nombreuses conseils sur mon projet professionnel. Je remercie notamment Lydie pour le partage des moments fun, et aussi de ceux moins bons, de nos thèses et Loic, pour son soutien professionnel et non.

Merci à mon petit bout d'Italie à Paris, Benedetta, Paola, Marta, Francesca, Miguel, Valentina, Valeria Ca., Valeria Co., Luca et à mes amies Dena, Elodie de m'avoir fait sentir chez moi ici.

Dans ces remerciements ne peuvent manquer mes amies Alice, Silvia et Michela. Les mathématiques nous ont rapproché mais beaucoup plus me lie intimement à vous.

Au terme de ce parcours, je remercie celles et ceux qui me sont chers et que j'ai quelque peu délaissés ces derniers mois pour achever cette thèse. Je suis redevable à mes parents, Antonietta et Piero, pour leur soutien moral et matériel et leur confiance indéfectible dans mes choix. Merci à mon frère Paolo de faire partie de ma vie. Merci aussi à mes oncles Roberta et Franco pour leur soutien constante et tenace tout le long de ma vie. Merci à mes cousines Nadia et Silvia d'être toujours là pour me souhaiter le bentornata a casa.

Merci enfin, les mots n'arrivent pas, à mon compagnon de rue Francesco de partager à present ma vie.

Contents

Introduction

This thesis is motivated by a challenging problem in anthropology and forensic science, that of tridimensional *facial reconstruction*. Facial reconstruction is about the following question: from the sole datum of a dry skull, are we able to reconstruct/estimate a virtual face that would trig the recognition of the subject?

This thesis is devoted to the conception, the development, and the analysis of mesh deformation strategies for shape modeling, processing and comparison -as shape matching and surface reconstruction- and, in a rather independent concern, for devising a robust computational method for facial reconstruction. The main contribution of the thesis is the design of a novel method for shape matching, borrowing techniques from the shape optimization context.

To better unravel the structure of the manuscript let us start with some considerations about the application motivating of this work.

Context of the work

This work has been realized in the context of the multi-disciplinary project FaciLe, gathering together maxillofacial surgeons, anthropologists, computer scientists and mathematicians from Sorbonne Universités. Facial reconstruction methods are demanded in forensic cases for enhancing the identification of deceased people when the skeletal remains is the sole information available for identification. Facial reconstruction is an *ill posed* problem, meaning that a facial shape is never uniquely determined from the underlying osseous structure. However, the strong correlation between a face and its skull is known since ancient times and has fascinated scholars over the ages [90]. It is only in the 19th century that facial reconstruction emerges as a common practice in forensic cases; due to the ethical, legal, social and psychological consequences of the verdict of a forensic identification, questions around the reliability of methods became of capital importance [118]. Traditional facial reconstructions are based on manual procedures -3d sculptures or 2d portraits- and accomplished by well-experienced forensic artists; the reliability of the result reposes on skills, anatomical knowledge and subjectivity of the practitioner. With the booming of computer science and medical imaging, many digital methods have been developed for a twofold purpose. First, digital methods eliminate the subjectivity of the reconstruction; then, taking advantage of the flexibility of the software one is able to test different hypoth-

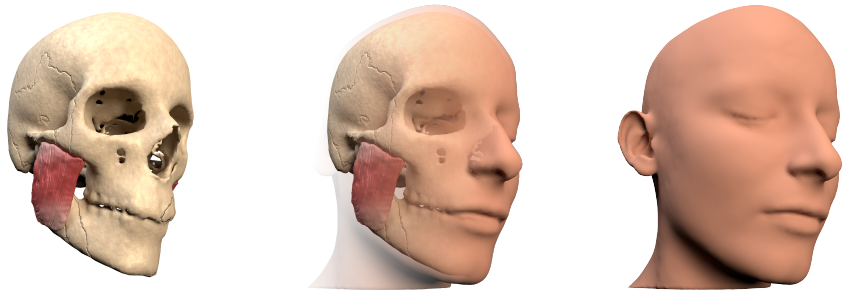
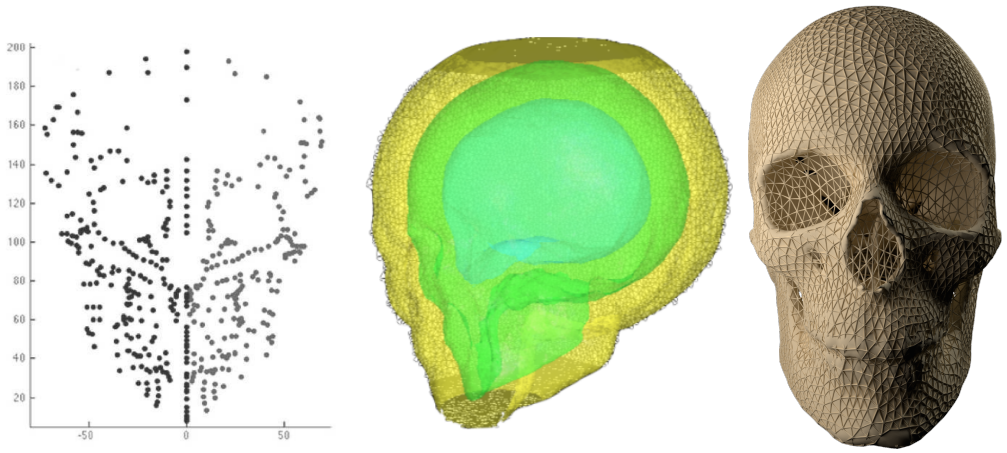


Figure 1: Goal of facial reconstruction: estimation of the facial shape from the underlying bone.

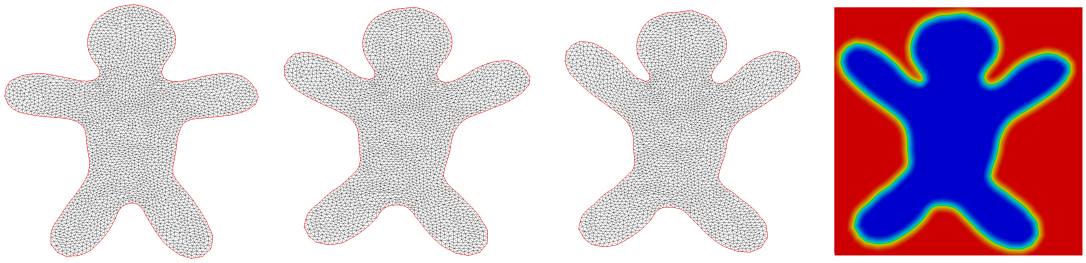
esis and propose multiple instances of the reconstruction. Most of actual digital methods combine statistical analysis with the observation of a database of coupled skulls and faces for learning the relations between face and skull. The quantitative analysis of shapes is studied in forensic anthropology using the techniques of geometric Morphometrics, which is based on the statistical analysis of $3d$ coordinates of *landmarks*, defined as anatomical loci that are the same in all the specimen under investigation [15]. Landmarks are important for their biological meaning, but it is difficult to translate their definition into geometrical rules. Then, their the definition and placement on the shape under investigation (bones, anatomical structures) is done manually following descriptive anatomical rules. Current facial reconstruction methods have much in common with morphometric analysis methods. In either manual and digital approaches, a set of anatomical landmarks is usually defined on the skull and corresponding face to account for the soft tissue variability [26, 118]. The correlation between the underlying bone structure and the outer facial shape is then studied by looking at the soft tissue thickness measured at landmarks. The reconstruction of the face coupled with the unknown skull is inferred by a statistical estimation of the facial landmarks and/or by a template deformation approach. However, the manual effort required for landmark-based methods is important and increase with the size of the database under examination, making difficult the comparison of a huge amount of data. One concern of this manuscript was to design a fully-automated procedure, suitable for processing a huge amount of data, using dense surfaces instead of landmarks. Facial reconstruction is *intrinsically* an interdisciplinary problem. With the advent of digital methods, forensic experts has started to interact with mathematicians and computer scientists for the development of robust tools for analysing and comparing complex $3d$ shapes. The main concern of project FaciLe was to combine a strong mathematical background, $3d$ modeling expertise and robust software implementation at the service of an interdisciplinary discussion about the subject of facial reconstruction. Based on techniques of continuous deformation as 'morphing' and 'warping', the proposed approach is integrated with anthropological assumptions and mechanical models. Our approach combines classical features -as the use of a head database of faces and skulls or the anthropological expertise for



(a) (b) (c)

Figure 2: Three ways of accounting for a the skull shape. (a) Semi landmarks, reprinted from [13]. (b) Implicit description. (c) Closed surface mesh model.

classifying the unknown skull- with mathematical and computational skills as Partial Differential Equations (PDE), numerical analysis and $3d$ geometric modeling. The whole process is landmark-free and relies on general shapes analysis tools instead of application-specific ones. The general pipeline of our approach for reconstructing the face coupled with an unknown skull can be summarized in few points. First, a craniofacial database of coupled skull and face surfaces is generated. Then, an examination of the unknown skull determines anthropological parameters like age, sex, ethnicity and, according to these parameters, some instances of coupled faces and skulls are selected among the database. For each selected individual, a $3d$ volumetric mask -which we will call craniofacial template- is generated for encoding the specific soft tissue variability. The reconstruction of the face coupled with the unknown skull is done by deforming the craniofacial template onto the unknown skull. This stage is accomplished in two steps. First, the inner surface of each mask (i.e. the bone surface) is mapped onto a digital copy of the unknown skull by a landmark-free elastic dense surface morphing. The considered mask is then 'elastically' transported onto the unknown skull for deforming the facial template according to the previously computed skull mapping. Doing so, each individual selected in the database generates an estimate of the unknown face. The set of facial shapes are eventually combined together for proposing -one or multiple- instances of the reconstructed facial shape. In a final stage, hairiness and texture are added to the reconstructed shape for enhancing lifelike appearance.



(a) (b) (c) (d)

Figure 3: $2d$ elastic morphing. (a) Template shape. (d) Signed distance function to the target shape. (b)-(c) Deformation of the template shape (arms raised) for aligning its boundary with the isolines of the signed distance function.

An optimization method for elastic shape matching

The key ingredient of our reconstruction method is an original shape matching technique, devised in collaboration with Maya de Buhan, Charles Dapogny and Pascal Frey. Beyond the specific task of our application, shape matching or morphing relates to the following task: given a couple of 'template' and 'target' shapes Ω_0 and Ω_T in $\mathbb{R}^d (d = 2, 3)$, find a method for mapping/morphing/matching Ω_0 onto Ω_T. Such a transformation may be used as a means to appraise how much Ω_0 and Ω_T differ from one another - for instance in shape retrieval, classification or recognition - or to achieve physically the transformation from Ω_0 to Ω_T (in shape registration or reconstruction). Shape matching plays a crucial role in most geometric modeling and shape analysis applications [111]; a huge interest has already been devoted to this topic among the mathematics and computer graphics communities [40, 12, 10, 7, 97, 70]. Our approach relies on a continuous optimal control problem. Under the sole assumption that the two shapes share the same topology, the desired mapping is achieved by minimizing an energy functional based on the distance between the two shapes. For achieving our goal we rely on the shape optimization setting. Shape optimization problems are optimization problems involving a functional $F(\Omega)$ depending on a domain Ω. The criterion $F(\Omega)$ is optimized by searching over a set of admissible shapes $\mathcal{O}$. Generally a shape functional depends on the domain in a complex way, for instance via the solution of a differential equation posed on the domain, making the optimization problem highly non trivial. We propose to measure the discrepancy between the 'template' shape Ω_0 and the 'target shape' Ω_T in terms of the following shape functional:

$$J(\Omega) = \int_{\Omega_0} d_{\Omega_T}(x) dx,$$

involving the *signed distance function* d_{Ω_T} to Ω_T. The signed distance function is a popular level set function of a domain; for each point x of the ambient space, this function returns the Euclidean distance $d(x, \partial\Omega_T)$ to the boundary $\partial\Omega_T$. The sign of

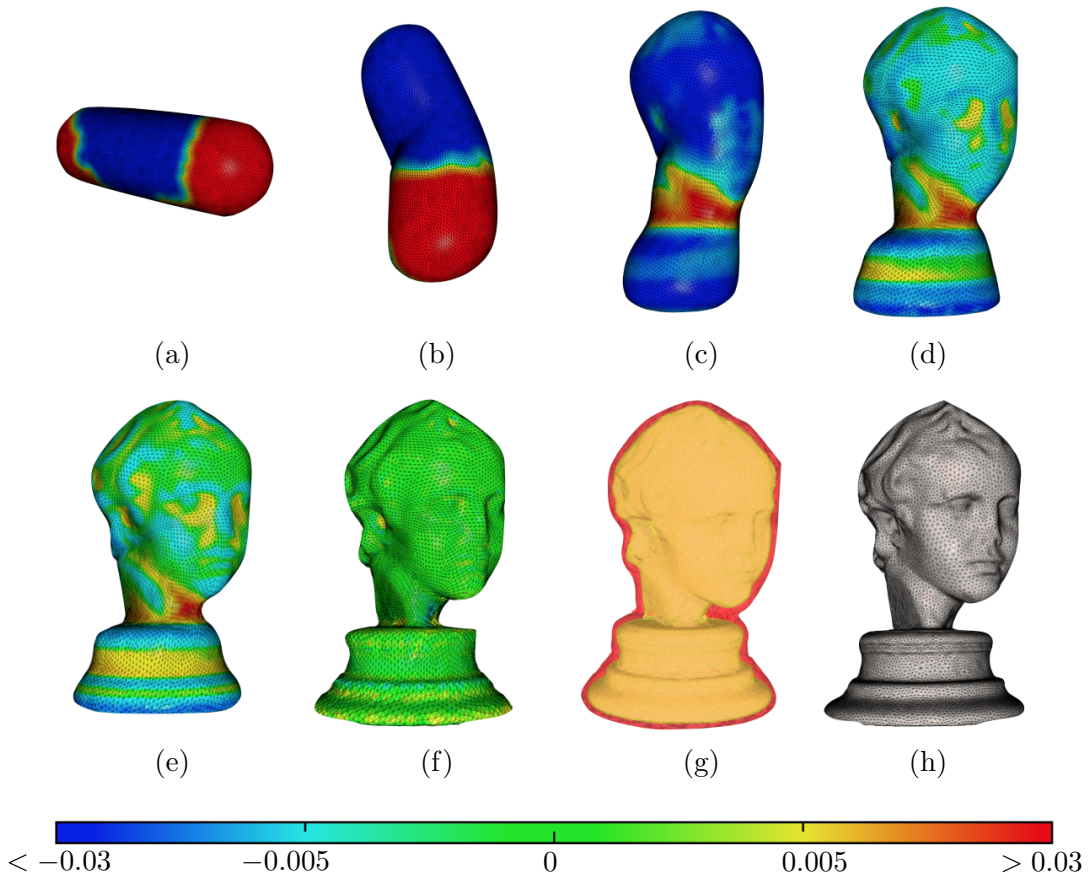


Figure 4: $3d$ elastic morphing of a capsule model into the *Eros* model. (a) Template shape. (g) Isovalues of the signed distance function to the target shape. From (b) to (f): deformation of the template shape for decreasing the values of J. (h) Mesh of the template shape.

the function is chosen negative if the point lies at the interior of the domain, positive otherwise. The isoline (resp. isosurface) $\{x \in \mathbb{R}^d, d_{\Omega_T}(x) = 0\}$ captures the boundary $\partial\Omega_T$ of the domain. For decreasing the values of J, the template shape has to retract in the region of spaces in which d_{Ω_T} is negative and to expand in those in which it is positive. Consider a shape $\Omega \subset \mathbb{R}^d$ and suppose $\Omega_T \subset \Omega$. In this case, it is beneficial for Ω to retract for decreasing the values of J. In contrast, when Ω lies at the interior of Ω_T, it will be advantageous for Ω to expand for gaining a negative contribution and decreasing the values of J. From the above considerations, we find out that the functional $J(\Omega)$ admits a unique global minimizer equal to the target shape Ω_T, which is a quite nice property for a shape optimization problem. The above existence property follows from the fact that the functional J is defined on the whole volume Ω_0. The global mapping between Ω_0 and Ω_T is achieved by an

iterative procedure devoted to the minimization of J, inferring at each step a descent direction for the functional J. The desired deformation is achieved numerically by deforming iteratively a mesh of the template shape into a computational mesh of the target shape. As far as the numerical setting is concerned, the template shape is described by means of a computational mesh, and the target shape is known only via the signed distance function to its boundary. We rely on a gradient descent algorithm for iteratively deforming the template shape. The optimality conditions associated to our shape optimization problem are calculated using the framework of the Hadamard's shape derivative [18, 1, 60]. Roughly speaking, variations of a domain Ω are parametrized in terms of vector fields, i.e. are considered under the form $\Omega_\theta = (I+\theta)(\Omega)$, where θ is a 'small' vector field which we will make precise in the following. Doing so, the definition of differentiability over a set of admissible shapes is translated into a problem of differentiation over a Banach space, which is equipped with the classical definitions of derivative. The iterative procedure devoted to the minimization of J produces a sequence $(\Omega_k)_{k=0,\dots}$ of shapes (and corresponding meshes $\mathcal{T}_k$), which are 'closer and closer' to Ω_T. At each iteration, the domain is deformed according to a a descent direction θ for the functional J. The descent direction θ is inferred from the expression of the shape derivative $J'(\Omega)$, so as to guarantee the strict inequality $J((I+\theta)\Omega)) < J(\Omega)$ for θ small enough. Therefore, the mesh is updated by pushing its vertices according to (a discrete version of) the descent direction θ. At this stage, two difficulties which are both numerical and theoretical come into play. First, a classical calculation shows that the 'natural' descent direction associated to the functional J makes sense only on the boundary of Ω; it has therefore to be extended to Ω as a whole so that it can be a guide for displacing the vertices of $\mathcal{T}$. Second, the mesh update procedure has to be achieved in a 'smart' way for avoiding inversion or stretching of elements, which can result in an invalid configuration. For encompassing both this difficulties, the 'natural' descent direction for J has to be *extended* and *regularized*. For achieving this goal, we designed a strategy based on the particular mechanical system of *linearized elasticity*. Linear elasticity equations account for the displacement of a material body which is clamped at a part of its boundary and subjected to body or surface loads, in the range of 'small' deformations [22]. The general idea is to replace the 'natural' descent direction θ associated to J with a well-suited elastic displacement u which (i) is an extension of θ; (ii) is defined on the whole shape Ω; (iii) owning to the mechanical features and regularizing effect of elliptic equations, is smoother than θ, and so more amenable for moving the mesh into a valid configuration. In practice, the space of admissible deformations is restrained by parametrizing with elastic displacements. For achieving this goal, at each iteration the elastic extension is computed by using the gradient of $\theta \mapsto J'(\Omega)(\theta)$ associated to the inner product inherited from the operator of linear elasticity. This strategy has proved extremely versatile in numerical practice, allowing to map shapes which are not necessarily close from one other. The elastic equations have been implemented in a Finite Element setting. Figures 3 and 4 show two examples of such a procedure.

A different -but close in essence- strategy consists in imposing a displacement field $u_{\Omega,g}$ generated by a surface load g acting on the boundary of the template domain

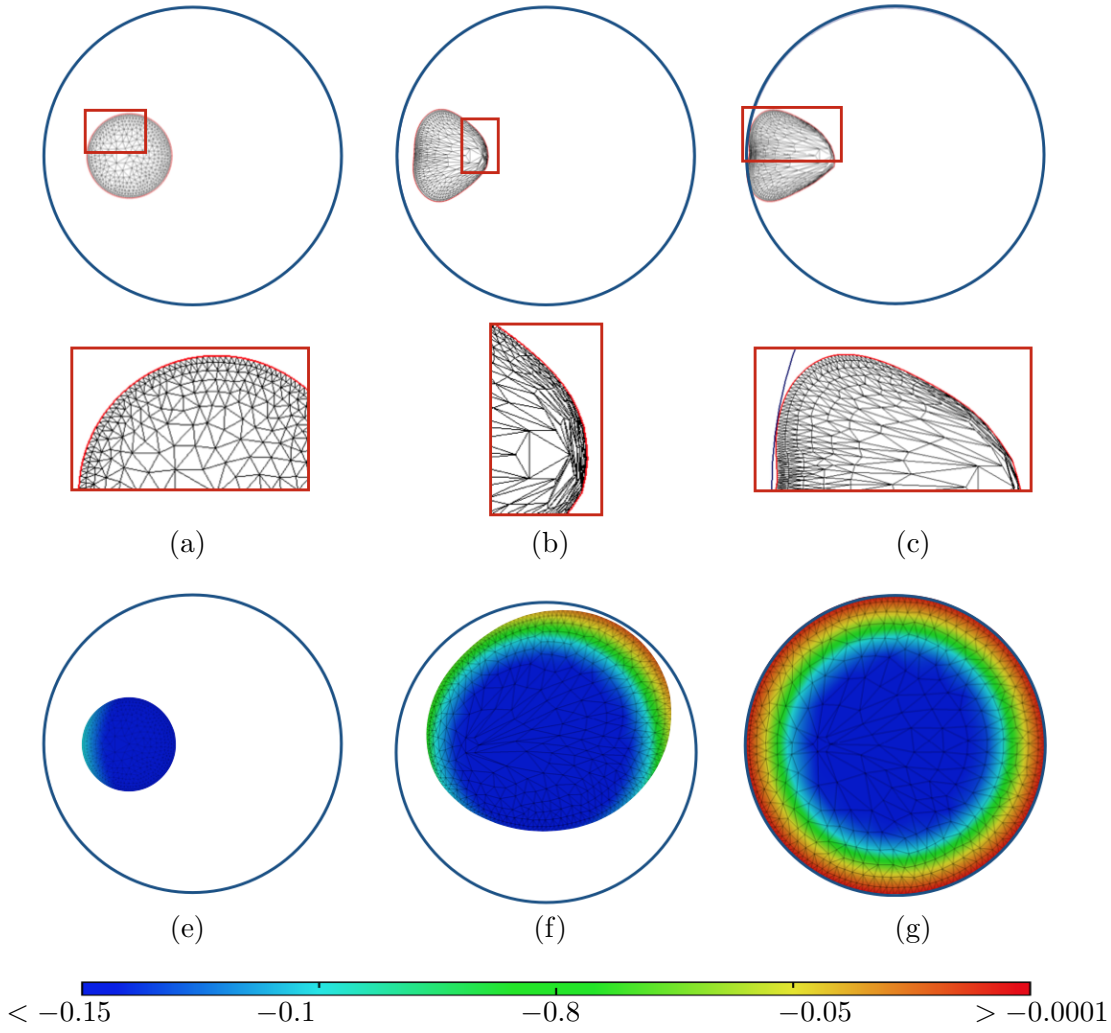


Figure 5: Toy example showing the advantages of volume shape matching over its boundary counterpart. (a)-(e) Computational mesh of the template shape (small disk) and boundary of the target shape (big disk in blue color). (b)-(c) Occurrence of singularities after few iterations when deforming the template shape according to the functional P. (f)-(g) Successful recovering of the target shape when the mesh deformation is driven by J.

Ω. A slight modification of J into a state-constrained functional allows to fulfill this requirement. In this context, the shape functional J is minimized over a set of admissible surface loads $\mathcal{O}_{ad}$. In this case, the computation of the descent direction for J needs to take into account the derivative of the state $u_{\Omega,g}$ with respect to the domain Ω. For rigorously assessing this calculation one has to refer to the notions of Eulerian and Lagrangian derivatives [1]. An alternative, very fast way for achieving a formal calculation relies on the use of Céa's fast derivation method [18]. The application of Céa's method leads to an explicit expression of the shape derivative in

terms of the state equation and a well-suited adjoint state. Despite the two numerical strategies have much in common, they revealed quite different numerical behaviors. In particular, for ensuring convergence of the overall optimization problem using the second strategy, one may suppose that target and template shapes are close enough, which is in practice a very limiting requirement.

Comparison with a boundary functional

The functional J has also been compared to its more classical surface counterpart, namely the functional of the domain:

$$P(\Omega_0) = \int_{\partial\Omega_0} \left(d(s, \partial\Omega_T)\right)^2 \, ds,$$

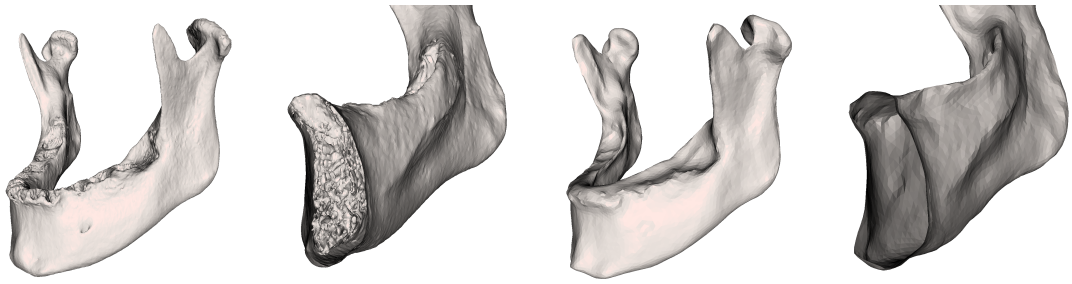
which depends on the euclidean distance function $d(\cdot, \partial\Omega_T)$ to $\partial\Omega_T$, evaluated at the boundary of Ω_0. The functional P is based on a more intuitive paradigm -aligning two shapes by aligning their boundaries- and has the advantage to be well-defined on surfaces instead of shapes. However some drawbacks make the functional P less suitable for numerical practice. First, the computational cost of each iteration is more involved when comparing with J, since the shape derivative of boundary functionals -even at first order- involves terms as the mean curvature of the boundary and the derivative of the integrand function. Second and perhaps most important aspect, for decreasing the values of P, a point lying on $\partial\Omega$ is forced to move to the closest point on $\partial\Omega_T$; this behavior can generate ambiguous configurations, leading to numerical singularities and preventing the convergence of the overall algorithm. See Figure 5 for a comparison between J and P on a $2d$ shape matching toy problem.

Around an existence problem in shape optimization

The functional J has proven useful in the different context of shape optimization applied to architectural design [32]. The main idea is to use J as a geometrical constrain in a structural optimization problem for imposing over the set of admissible shapes a requirement of the type 'stay close to a specific design Ω_T'. This approach has lead to the consideration of the following functional of the domain:

$$S_\gamma(\Omega) = \gamma C(\Omega) + (1-\gamma) J(\Omega),$$

which is minimized over the set $\mathcal{O}_V$ of admissible shapes sharing a fixed volume $0 < V < 1$. In the above formula, $C(\Omega)$ account for the *elastic compliance* of a structure which is subjected to a fixed, used defined surface or volume load, $J(\Omega)$ is the above defined discrepancy functional depending on a fixed target shape Ω_T and $\gamma \in [0, 1]$ is a weighting parameter. The compliance is defined as the elastic energy stored during the deformation, or equivalently as the work of external loads [1]. The functional S_γ induces the following interesting theoretical problem: there exist values



(a) (b)

Figure 6: (a) Mandible mesh model issued from segmentation of medical images. (b) Closed surface model of the mandibule generated with our warping algorithm.

of the weighting parameter γ for which the problem of minimizing S_γ over the set $\mathcal{O}_V$ is well-posed? When $\gamma = 0$, S_γ equals J and the problem admits a unique minimizer Ω_T. Oppositely, when $\gamma = 1$ the problem of minimizing S_γ turns into a compliance minimization problem under volume constraint, which is a well known example of ill-posed shape optimization problem. It can be shown that the optimum design for such a problem is a density function instead of a classical black/white domain. The non existence of classical solutions for this problem is due to the occurrence of homogenization problems [1]. It is well known that under additional geometrical, topological or regularity constraints -for instance imposing a finite perimeter or an upper bound over the number of connected components- the compliance problem admits a (regular) minimizer [60]. The additional constraints add extra compactness properties to the functional, with respect to some topology which we do not make precise now. An analysis of the optimality conditions leads to a characterization of the optima of the functional S_γ. However, this characterization does not suffice for concluding about a (non) existence result. At the moment of writing, the problem about the existence for $0 < \gamma < 1$ is still an open question. We discuss a state of the art of techniques for getting existence results for shape optimization problems and also perspectives for the study of the functional.

A warping algorithm for reconstruction of closed surface models

From both theoretical and numerical point of view, for using our shape matching method we need the reference and target shape to be bounded by a closed, orientable surface. This requirement allows us to define unambiguously interior and exterior of the considered shape. This representation allows to infer a topological structure to the considered shapes; such a step is needed for generating a tetrahedral mesh of the template mesh and for attributing a sign to the signed distance function to the target shape. Also, most CAD and engineering applications are based on this paradigm. However, most real-life models does not fulfill the above requirement, either because the geometry of the object is not covered by the above description or because of

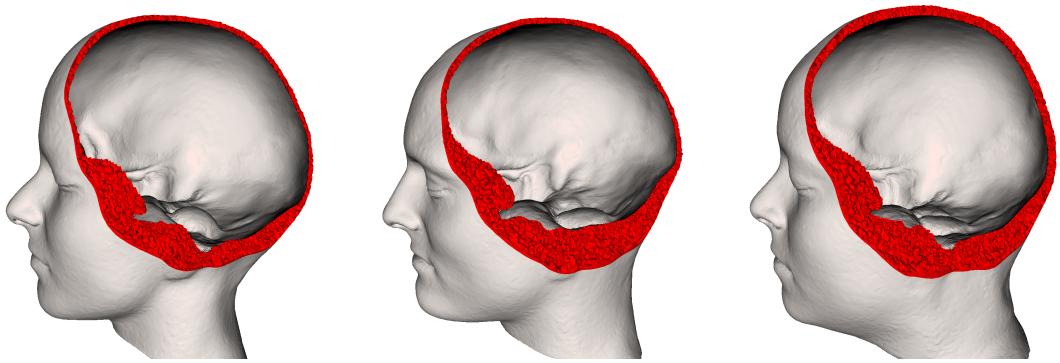
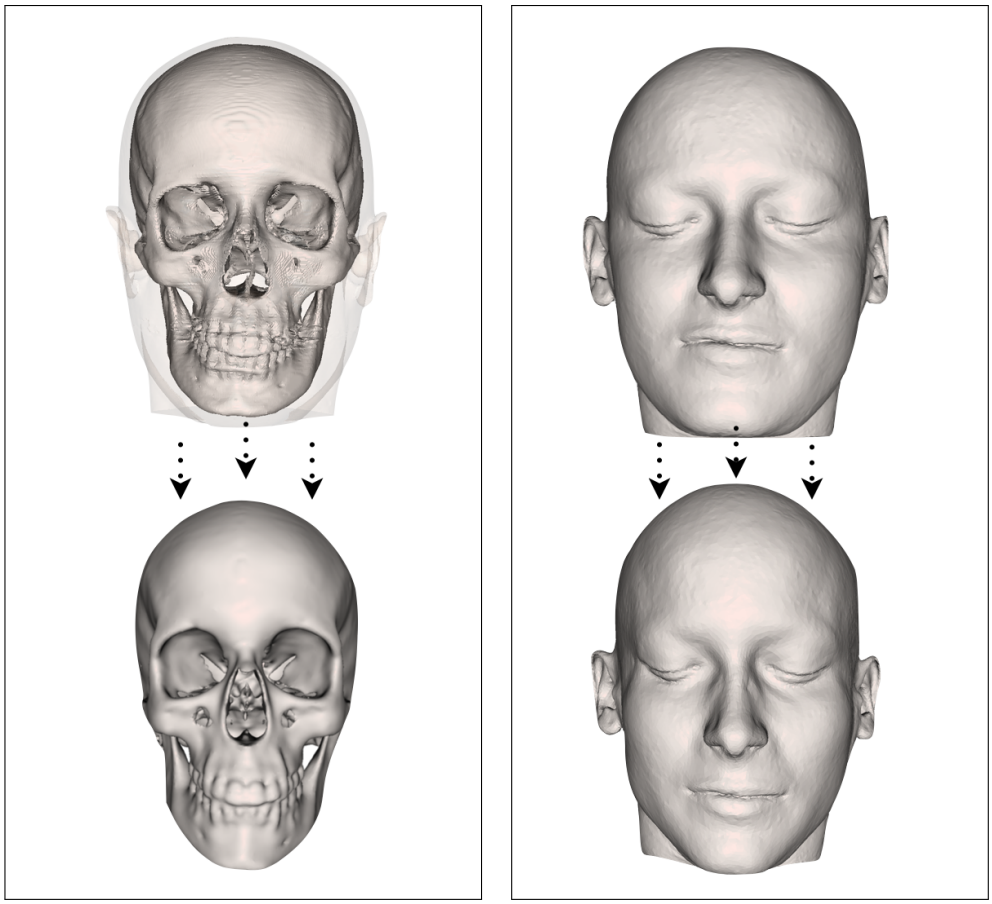


Figure 7: Three craniofacial templates.

errors occurred during the generating of the $3d$ model [39]. In the context of our specific application, we faced with this issue when it came to study the human skull, which has by nature a quite complex, non closed structure. For circumventing this difficulty, we developed a method for generating a closed surface mesh model of an initially non-closed source mesh model. The method relies on an original PDE-based mesh evolution technique. A template shape is iteratively deformed, producing a sequence of shapes that get 'closer and closer' (with respect to the Hausdorff distance) to the source triangulation. The topology of the template shape is kept unchanged during the deformation. The implementation of this technique, again, make use of the linear elasticity equations. The Finite Elements implementation of the algorithm has been tackled using two different strategies. In particular, tetrahedral and shell finite elements are considered. First, we considered the case in which the template shape to be deformed is given by a thick $3d$ membrane. This choice lead to the employment of the $3d$ liner elasticity equations using classical tetrahedral elements. Next, we consider shell finite elements adapted to the geometry of a surface. The considered finite elements depend on the two surface coordinates and a third component given by the normal to the surface [9, 20]. The proposed method can be also used for example for generating a *valid* mesh from invalid data. See Figure 6 for an example of such a procedure.

Back to the application

To end this introductory discussion, let us return focusing on our facial reconstruction problem. Our elastic matching method has been used at several stages of the facial reconstruction pipeline. First, the method is used to map the skull items into each others to learn about their similarities. The vicinity between a mapped template shape and and the target shape can be fixed up to a application-prescribed tolerance. The tolerance can be determined in our context according to the error allowed during the segmentation process. This requirement is achieved numerically by adjusting the size of the mesh of the template shape. Note that with our approach we



(a) (b)

Figure 8: (a) Deformation of a skull item onto the unknown skull. (b) Deformed craniofacial template under the effects of skull changes.

are able to generate computational meshes from invalid data, starting from the datum of the signed distance function to the shape. Moreover, by systematically deforming a reference mesh into each item of the database (face or skull) we generate a set of computational meshes enjoying the same number of elements and connectivity, that are linked each other by trivially composing their associated deformation functions. This property has revealed useful when constructing the craniofacial templates. This step is achieved numerically by filling with a a tetrahedral mesh the region of space between the skull and the face of each individual (see Figure 7). The generated 3d mask is then deformed and adapted to the unknown skull again in the context of linearized elasticity. The deformation is guided by the mapping between the skull item and the underlying skull. Under the effect of boundary changes, each facial item is mapped onto an estimate of the unknown face. See Figure 8 for an example of such a procedure. Finally, from the collection of deformed faces we are able to compute average shapes taking advantage of the point-to-point correspondence inferred from

the matching procedure.

Organization of the manuscript

The manuscript is organized in two parts, each one composed of four chapters. The first part is devoted to the introduction of the mathematical strategies, when the latter focuses on the facial reconstruction problem. By sacrificing mathematical and computational details, the second part can be read independently of the first.

Part I

Chapter 1
This chapter is bibliographical and is meant to give a very introductory survey about notions which are used in the rest of the manuscript. First, we will start with introducing shape optimization, general definitions and the theoretical background that we need for defining the notion of shape derivative of a state constrained functional. We present *en passant* an alternative way of measuring the shape sensitivity: the topological derivative. After introducing shape optimization fundamentals, we discuss informally about possible choices for describing shapes. In particular, we introduce the signed distance function -and in general the notion of level set function to a domain- and a brief recall of the differentiability properties of this function. Last but not least, the Chapter ends with an introduction to the linear elasticity setting. The equations of linear elasticity are recalled and, for the sake of completeness, a survey of existence, well-posedness, regularity of these equations is provided.

Chapter 2
In this chapter we describe our optimization method for elastic shape matching. After presenting a short, non exhaustive survey about the state of the art of shape matching techniques, we introduce our functional J based on the signed distance function. The general workflow of a descent algorithm for shape optimization is provided together with the calculation of the shape derivative of our functional. Theoretical and numerical aspects of the method are presented, giving to the reader the sufficient technical background for implementing the method. Several numerical examples are provided for showing efficiency and limits of the method. The boundary functional P is discussed as well, and some examples are proposed to show the limitations in numerical practice.

Chapter 3
This Chapter is more theoretical and focuses on the non existence of optimal shapes for the functional S_γ. Some well-known examples of non existence of optimal shapes for classical shape optimization problems are provided, as well as a survey about existence results achieved by adding extra requirements on the set of admissible shapes. The Chapter ends with an analysis of the optimality

conditions associated to the functional S_γ and perspectives for the study of the open existence problem.

Chapter 4
This chapter provides both theoretical and computational aspects of the warping algorithm used for generating a close surface mesh model from an initial, arbitrary source triangulation. The Finite Elements implementations of the model using tetrahedral and shell elements is discussed. Also, we present several numerical examples.

Part II

Chapter 5
This chapter is bibliographical and provides a quick state of the art about currents methods for facial reconstruction. After briefly recalling some historical circumstances, we describe the general workflow of a digital method.

Chapter 6
In this chapter we deal with the process of acquisition of our database of skulls and faces. In particular, we describe how we obtain the $3d$ geometrical model of the full head (skulls, faces, muscles) from medical images. This stage contains the following steps: segmentation of medical images, $3d$ reconstruction of a mesh model from the binary images, remeshing of the ill-shaped, oversampled model.

Chapter 7
This chapter describes the pipeline of our method for facial reconstruction, using the tools introduced in the first part of the manuscript. A detailed, user-friendly description of our method is provided; this portrayal is deliberately not technical.

Chapter 8
This chapter contains a gallery of results obtained with the proposed method and a discussion about perspectives and openings of the study. For the need of benchmarking, the proposed reconstructions are generated by removing the individuals under investigation from the craniofacial database. This procedure allow us to compare the estimated face with the original one.

Part of the work of this manuscript has been published into the journal article:

An optimization method for elastic shape matching, M. De Buhan, C. Dapogny, P. Frey, C. Nardoni, C.R. Acd. Sci., Paris, Sèrie I, 2016.

The following paper has been submitted:

A mesh deformation based approach for digital facial reconstruction, M. De Buhan, C. Nardoni.

Part I

Mesh deformation, shape optimization and linear elasticity primer

Chapter 1

Preliminary tools

Contents

1.1 Introduction

The aim of this chapter is to introduce some mathematical background which will be extensively employed in the rest of the manuscript. First, we will start with recalling some notions from shape optimization theory. In particular in Section 1.2 we will introduce the crucial notion of *shape derivative*, which allows for studying the sensitivity of a functional which is defined over a set of admissible shapes. Moreover, we will discuss about the computation of the shape derivative of functions which depend on a domain via a state equation, achieved for instance as the solution of a partial differential equation defined on the domain. Sections 1.3 and 1.4 discuss about two alternative ways for describing shapes. First, we will introduce the level set (or implicit) description, and in particular one peculiar instance of level set function to a domain, namely the *signed distance function*. Second, we will deal with the explicit description in which the shape under consideration is equipped with a *mesh*. Eventually, Section 1.5 deals with a particular mechanical system which will be ubiquitous in the manuscript, that of linear elasticity.

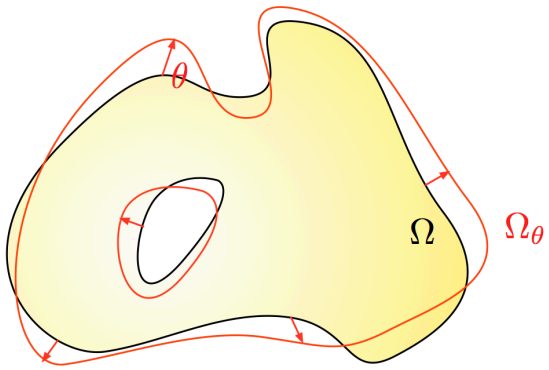


Figure 1: Variation Ω_θ of a shape Ω according to Hadamard's method.

1.2 A quick introduction to shape optimization

Generally speaking, shape optimization is about optimizing a cost functional $J(\Omega)$ -or objective functional- over a set of admissible shapes $\mathcal{O}_{ad}$. A generic shape optimization problem reads as:

$$\min_{\Omega \in \mathcal{O}_{ad}} F(\Omega), \text{ or } \max_{\Omega \in \mathcal{O}_{ad}} F(\Omega).$$

Shape optimization is a booming area with applications ranging from aerodynamic, fluid, structural optimization. When the optimization problem is casted over a set $\mathcal{O}_{ad}$ of shapes, some additional difficulties come into play, from both theoretical and numerical point of view. First, one needs a way for describing admissible variations over the space $\mathcal{O}_{ad}$ and a notion of differentiation over the space $\mathcal{O}_{ad}$, which is needed for inferring optimality conditions. Moreover, in most problems the dependency of the objective functional on the domain is highly non-trivial; the functional J may depends on geometric properties of the domain (i.e. curvature, normals, thickness) but also on a state u_Ω - achieved as the solution of a PDE posed on Ω- and its derivatives. In this case we have to deal with the derivation of the state u_Ω under consideration with respect to the domain.

1.2.1 Shape sensitivity analysis: Hadamard's shape derivative

Several notions of differentiation with respect to the domain are available in the literature. Here we refer to the Hadamard's boundary variation method (see e.g. [1, 60, 81]). The general idea is to establish a correspondence between variations of the domain and diffeomorphisms of $\mathbb{R}^d$. Thus, variations of a given shape Ω are considered under the form:

$$\Omega_\theta = (I + \theta)(\Omega),$$

where $\theta : \mathbb{R}^d \mapsto \mathbb{R}^d$ is a 'small' diffeomorphism. Indeed, each admissible variation Ω_θ of Ω is parametrized in terms of a transformation of the form $I + \theta$, which remains 'close' to the identity. The admissible vector field θ is sought among the Banach space $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ of bounded and Lipschitz functions endowed with the norm:

$$||\theta||_{W^{1,\infty}(\mathbb{R}^d,\mathbb{R}^d)} := ||\theta||_{L^\infty(\mathbb{R}^d)^d} + ||\nabla\theta||_{L^\infty(\mathbb{R}^d)^{d\times d}}, \forall\theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d).$$

The convenience of the choice of this space follows from the following results (from [1]).

Lemma 1.2.1. *For any $\theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ such that $||\theta||_{W^{1,\infty}(\mathbb{R}^d,\mathbb{R}^d)} \leq 1$, the application $I + \theta$ is a Lipschitz homeomophism with Lipschitz inverse.*

Using the correspondence between vector fields and functions, the Hadamard shape derivative is defined in terms of the classical derivative over a functional space.

Definition 1.2.1. *A function $F(\Omega)$ of the domain is said to be shape differentiable at Ω if the mapping $\theta \mapsto F(\Omega_\theta)$, from $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ into $\mathbb{R}$, is Fréchet differentiable at $\theta = 0$. The associated Fréchet differential is denoted as $\theta \mapsto F'(\Omega)(\theta)$ and called the shape derivative of F; the following expansion then holds:*

$$F(\Omega_\theta) = F(\Omega) + F'(\Omega)(\theta) + o(\theta), \text{ where } \frac{|o(\theta)|}{||\theta||_{W^{1,\infty}(\mathbb{R}^d,\mathbb{R}^d)}} \xrightarrow{\theta\to 0} 0.$$

Following the same construction one can define weaker notions of shape differentiability. So on, a functional $F(\Omega)$ of the domain is said to admit a directional (or Gâteaux) derivative $F'(\Omega;\theta)$ at Ω in the direction θ if $\theta \mapsto F(\Omega_\theta)$ admits a Gâteaux derivative at Ω in the direction θ. In this case we pose:

$$F'(\Omega;\theta) = \lim_{t\to 0} \frac{F\left((I + t\theta)\right)(\Omega) - F(\Omega)}{t}.$$

The choice of $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ as underlying Banach space guarantees the requirement $(I + \theta)$ to be invertible with Lipschitz inverse. In most applications it is in practice more convenient to work with continuous functions, considering for $k > 0$ the Banach space $\mathcal{C}^{k,\infty}(\mathbb{R}^d, \mathbb{R}^d) = \mathcal{C}^k(\mathbb{R}^d, \mathbb{R}^d) \cap W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ equipped with the norm:

$$||\theta||_{C^{k,\infty}(\mathbb{R}^d,\mathbb{R}^d)} := \sum_{l=0}^{k} \sup_{\alpha\in\mathbb{N}^d, |\alpha|=l} \left|\left|\frac{\partial\theta}{\partial x^\alpha}\right|\right|_{L^\infty(\mathbb{R}^d)^d}, \forall\theta \in \mathcal{C}^{k,\infty}(\mathbb{R}^d, \mathbb{R}^d).$$

The application $(I + \theta)$ results to be a C^k diffeomorphism and the shape derivative is defined in terms of the Fréchet derivative at $\theta = 0$ of $\theta \mapsto F(\Omega_\theta)$ in the considered Banach space $\mathcal{C}^{k,\infty}(\mathbb{R}^d, \mathbb{R}^d)$. However, when the context is not ambiguous the two definitions of shape derivative are referred with the same name. A very important property of the shape derivative is that the form $\theta \mapsto F'(\Omega)(\theta)$ (at least at first order) does not depends on the tangential component of θ on $\partial\Omega$. More precisely we have the following statement (see [1]).

Proposition 1.2.1. *Let Ω be a bounded, Lipschitz domain in $\mathbb{R}^d$. Let F be a functional differentiable at Ω. Let $\theta \mapsto F(\Omega_\theta)$ be of class $\mathcal{C}^1$. If $\theta \cdot n = 0$, n standing for the outer normal to $\partial\Omega$, then we have:*

$$\forall \theta \in \mathcal{C}^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d), F'(\Omega)(\theta) = 0.$$

As a consequence, for each pair θ_1, θ_2 belonging to $C^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ and verifying the equality $\theta_1 \cdot n = \theta_2 \cdot n$ on $\partial\Omega$, then we have: $F'(\Omega)(\theta_1) = F'(\Omega)(\theta_2)$. The following two theorems (see [1, 60]) deal with the expression of the shape derivative for functionals defined over a whole domain or on its boundary.

Theorem 1.2.1. *Let $\Omega \subset \mathbb{R}^d$ a bounded, Lipschitz domain. Let f be in $W^{1,1}(\mathbb{R}^d, \mathbb{R}^d)$. The functional*

$$F(\Omega) = \int_\Omega f(x)\, dx$$

is shape differentiable at Ω, and the associated shape derivative reads:

$$\forall \theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d), \;\; F'(\Omega)(\theta) = \int_{\partial\Omega} f\, \theta \cdot n\, ds,$$

where n stands the normal vector pointing outward to $\partial\Omega$.

Theorem 1.2.2. *Let $\Omega \subset \mathbb{R}^d$ a C^2 domain. Let f be in $W^{2,1}(\mathbb{R}^d, \mathbb{R}^d)$. The functional*

$$F(\Omega) = \int_{\partial\Omega} f(x)\, dx$$

is shape differentiable at Ω, and the associated shape derivative reads:

$$\forall \theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d), \;\; F'(\Omega)(\theta) = \int_{\partial\Omega} \left(\frac{\partial f}{\partial n} + \kappa f \right) \theta \cdot n\, ds,$$

where $\kappa = \;$ div $n\;$ stands for the mean curvature of $\partial\Omega$.

1.2.2 Shape derivative of state-constrained functionals

Let us now focus on shape optimization problems depending on the domain via a state u_Ω which itself depends on the domain. We are here interested in the case in which the state u_Ω is obtained as the solution of a partial differential equation defined on Ω. For calculating the shape derivative of such functionals we are confronted with a new difficulty, namely the computation of the derivative of the function $u_\Omega(x) = u(\Omega, x)$ with respect to Ω. This is not a straightforward task and we give here a general outline on the topic. For further references and examples see [1, 60, 30]. The most intuitive way to address the problem leads to the notion of Eulerian derivative. Consider a point $x \in \Omega$ and let $\Omega_\theta = (I + \theta)(\Omega)$ be the domain transported through a displacement field $\theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$. Assume at first that $x \in \Omega \cap \Omega_\theta$,which is a reasonable requirement if x lies far enough from $\partial\Omega$. Then one aims to define the

Eulerian derivative as the derivative of the application $\theta \mapsto u_{\Omega_\theta}(x)$, i.e. the linear form $l(u)$ verifying:

$$u(\Omega_\theta, x) = u(\Omega, x) + l(u)\theta + o(\theta), \text{ where } \frac{|o(\theta)|}{||\theta||_{W^{1,\infty}(\mathbb{R}^d,\mathbb{R}^d)}} \xrightarrow{\theta\to 0} 0.$$

The difficulty in making rigorous the above definition is that $u_{\Omega_\theta}(x)$ could not be defined on $\partial\Omega$. For circumventing this difficulty we needs to rely on the notion of Lagrangian derivative. Consider the application $\Omega \mapsto u_\Omega$ and for all $\theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ define the 'transported' function $\bar{u}(\theta) : W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d) \mapsto H^1(\Omega)$ as follows:

$$\bar{u}(\theta) = u_{\Omega_\theta} \circ (I + \theta).$$

Definition 1.2.2. *(Lagrangian derivative) The function u of the domain Ω is said to admit a Lagrangian derivative $\dot{u}(\Omega)(\theta)$ at Ω in the direction θ if the function $\bar{u}(\theta) : W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d) \mapsto H^1(\Omega)$ is differentiable at $\theta = 0$.*

From the definition of the Lagrangian derivative we can infer a rigorous definition of the Eulerian derivative by applying the chain rule.

Definition 1.2.3. *(Eulerian derivative) The function u of the domain Ω is said to admit an Eulerian derivative at Ω in the direction θ if it admits a Lagrangian derivative at Ω in the direction θ and, provided $\nabla u(\Omega) \cdot \theta \in H^1(\Omega)$, the Eulerian derivative $u'(\Omega)(\theta)$ is defined as:*

$$u'(\Omega)(\theta) = \dot{u}(\Omega)(\theta) - \nabla u(\Omega) \cdot \theta. \tag{1.2.1}$$

The computation of an explicit expression of the Lagrangian and Eulerian derivatives is needed for the derivation of functionals of the form:

$$F(\Omega) = \int_\Omega f(u_\Omega) dx,$$

where u_Ω is the solution of a variational problem posed on Ω. The general procedure for carrying out the derivative $F(\Omega)$ involves the following steps:

- write the variational equation satisfied by u_{Ω_θ} and by a changing of variables transform it on an equation on $\bar{u}(\theta)$ posed on the fixed domain Ω;
- compute the Lagrangian derivative of u;
- compute the Eulerian derivative from (1.2.1) for deriving an expression of $F'(\theta)$;
- introduce an appropriate adjoint problem for eliminating the dependency of $F'(\theta)$ on $u'(\theta)$.

The above sketch does not precise the functional spaces involved because they depend on the specific equation at hand. Despite the above method is the rigorous way

to verify the differentiability of a shape functional, the computation of the material derivative is very delicate and involved in terms of computations. Moreover, by using the adjoint problem we end up with an expression of the shape derivative not depending on $u'(\theta)$. A faster alternative method for the computation of the shape derivative is the so-called Céa's fast derivation method [18], which we describe in the following. Despite the method is merely formal because it supposes the differentiability of the mapping $\Omega \mapsto u_\Omega$, it can be justified in a rigorous way in some cases. Let V be a normed vector space, $a_{\Omega,V} : V \times V \mapsto \mathbb{R}$ a symmetric, continuous and coercive bilinear form and $l_{\Omega,V} : V \mapsto \mathbb{R}$ a continuous linear form. Let us introduce the state u_Ω as the unique solution of the variational problem:

$$\forall v \in V,\ \ a_{\Omega,V}(u, v) = l_{\Omega,V}(v).$$

Let us consider a cost functional of the form:

$$F(\Omega, u) = \int_\Omega E(u)\, dx + \int_{\partial\Omega} G(u)\, ds,$$

where E, G are integrable functions depending on u. Pose $F(\Omega) := F(\Omega, u_\Omega)$. The key idea is to introduce a Lagrangian functional $\mathcal{L}$ which integrates the state u as a constraint. Thus, define $\mathcal{L} : \mathcal{O}_{ad} \times V \times V \mapsto \mathbb{R}$ as follows:

$$\mathcal{L}(\Omega, u, p) = F(\Omega, u) + a_{\Omega,V}(u, p) - l_{\Omega,V}(p).$$

The partial derivative of $\mathcal{L}(\Omega, u, p)$ with respect to p reads:

$$\forall v \in V,\ \ \frac{\partial \mathcal{L}}{\partial p}(\Omega, u, p)(v) = a_{\Omega,V}(u, v) - l_{\Omega,V}(v) = 0, \tag{1.2.2}$$

whenever $u = u_\Omega$. By equating to zero the partial derivative of $\mathcal{L}(\Omega, u, p)$ with respect to u we obtain the equation satisfied by the adjoint state p_Ω:

$$\forall v \in V,\ \ \frac{\partial \mathcal{L}}{\partial u}(\Omega, u, p)(v) = 0. \tag{1.2.3}$$

Eventually supposing enough regularity on u and using (1.2.2) and (1.2.3) we end up with the equality:

$$\forall v \in V,\ \ F'(\Omega)(v) = \frac{\partial \mathcal{L}}{\partial \Omega}(\Omega, u_\Omega, p_\Omega)(v).$$

To fix ideas, let us consider an example taken from [18]. Take $V = H^1(\Omega)$ and $u_\Omega \in V$ solution of the problem

$$\begin{cases} -\Delta u_\Omega = f, & \text{in } \Omega, \\ u_\Omega = u_0, & \text{on } \partial\Omega, \end{cases}$$

where $f \in L^2(\Omega)$ is the source term and $u_0 \in V$ is the boundary condition. The

functional under consideration is a least square criterion:

$$F(\Omega, u_\Omega) = \frac{1}{2}\int_\Omega |u - u_T|^2,\ dx,$$

where $u_T \in V$ is a target function. The associated Lagrangian reads:

$$\mathcal{L}(\Omega, u, p, \lambda) = \frac{1}{2}\int_\Omega |u - u_T|^2,\ dx + \int_\Omega (-\Delta u_\Omega - f)p\ dx + \int_{\partial\Omega} (u - u_0)\lambda\ dx,$$

in which we introduced the additional parameter λ for finding the boundary conditions satisfied by the adjoint state. The adjoint state is determined by choosing p and λ such that the following identity holds:

$$\int_{\partial\Omega} (u - u_0)v\ dx + \int_\Omega (-\Delta u)p\ dx + \int_{\partial\Omega} v\lambda\ ds = 0.$$

A Green formula yields:

$$\int_{\partial\Omega} (u - u_0)v\ dx + \int_\Omega \frac{\partial v}{\partial n}p\ dx = \int_{\partial\Omega} v(-\Delta p)\ dx + \int_{\partial\Omega} \frac{\partial p}{\partial n}v\ ds = 0.$$

Therefore p_Ω is the solution of the following problem:

$$\begin{cases} -\Delta p_\Omega = -u_\Omega - u_0, & \text{in } \Omega, \\ \nabla p_\Omega = 0, & \text{on } \partial\Omega, \end{cases}$$

and accordingly $\lambda_\Omega = -\frac{\partial p_\Omega}{\partial n}$. Using the expressions of u_Ω, p_Ω and λ_Ω the shape derivative of F is given by Theorems 1.2.1 and 1.2.2. We end up with:

$$\begin{aligned} F'(\Omega)(v) &= \frac{\partial \mathcal{L}}{\partial \Omega}(\Omega, u_\Omega, p_\Omega, \lambda_\Omega)(v) \\ &= \int_{\partial\Omega} \big(\frac{1}{2}|u_\Omega - u_0|^2 + (-\Delta u_\Omega - f)p + \lambda\frac{\partial(u_\Omega - u_0)}{\partial n} + (u_\Omega - u_0)\lambda\kappa\big)v \cdot n\ ds \\ &= \int_{\partial\Omega} \big(\frac{1}{2}|u_\Omega - u_0|^2 - \frac{\partial p}{\partial n} \cdot \frac{\partial(u_\Omega - u_0)}{\partial n}\big)v \cdot n\ ds. \end{aligned} \tag{1.2.4}$$

1.2.3 Shape sensitivity analysis: the topological gradient

The main idea in Hadamard's boundary variation method is to work with a fixed domain and consider variations obtained through a bi-Lipschitz map between this domain and the modified one. This approach cannot handle topological changes. We need thus to consider a new notion of variation of a domain. The main idea is to evaluate the variation of the objective functional when a small hole is cut into the admissible shape. Obviously, this type of variation of the domain is not covered by the

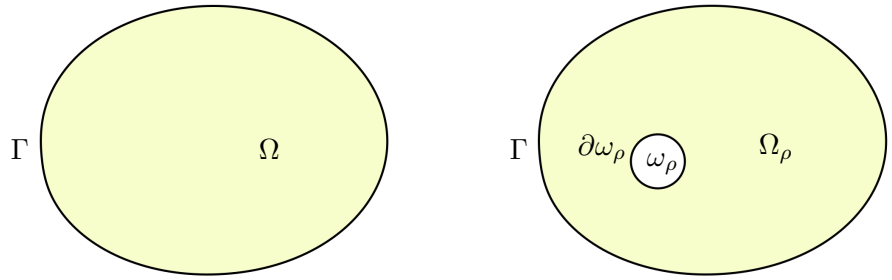


Figure 2: Initial domain before (left) and after (right) nucleating an hole.

Hadamard method, and it yields a new concept of topological derivative. Basically it indicates where a small hole should be put in the shape. Let $\Omega \subset \mathbb{R}^d$ be a open, bounded domain. For a given x_0 we consider the open set $\omega_\rho = x_0 + \rho\omega$, where $\omega \subset \mathbb{R}^d$ is a open set containing the origin. We define the perforated domain Ω_ρ obtained by removing ω_ρ from Ω:

$$\Omega_\rho = \Omega \setminus \overline{\omega_\rho},$$

and we are interested in the behavior of $F(\Omega_\rho)$ as $\rho \to 0$.

Definition 1.2.4. *A functional of the domain $F(\Omega)$ admits a topological derivative $F'_T(x)$ at a point x if there exists a continuous function $f : \mathbb{R} \to \mathbb{R}$ such that the following expansion holds in the neighborhood of $\rho = 0$:*

$$F(\Omega_\rho) = F(\Omega) + F'_T(x) f(\rho) + o(\rho),$$

with $\lim_{\rho\to 0} f(\rho) = 0, f(\rho) > 0$.

The topological sensitivity $J'_T(x)$ indicates where a small hole should be located in the shape. It follows from the above definition that $F(\Omega_\rho) < F(\Omega)$ for ρ small if $F'_T(x) < 0$.

1.3 The signed distance function

The key idea with implicit modeling is to represent the boundary of a shape as the zero level set of an embedding implicit function defined in the whole ambient space. Level set functions have proven useful in modeling deformations of solid objects, representation of medical data, dealing with topological changes by undergoing easily splitting of merging. A shape defined by a close boundary allows the definition of a level set function to its boundary encoding inside/outside informations. The definition of a level set function associated to a domain is reported below.

Definition 1.3.1. *Let $\Omega \subset \mathbb{R}^d$ be a bounded domain. A level set function of Ω is*

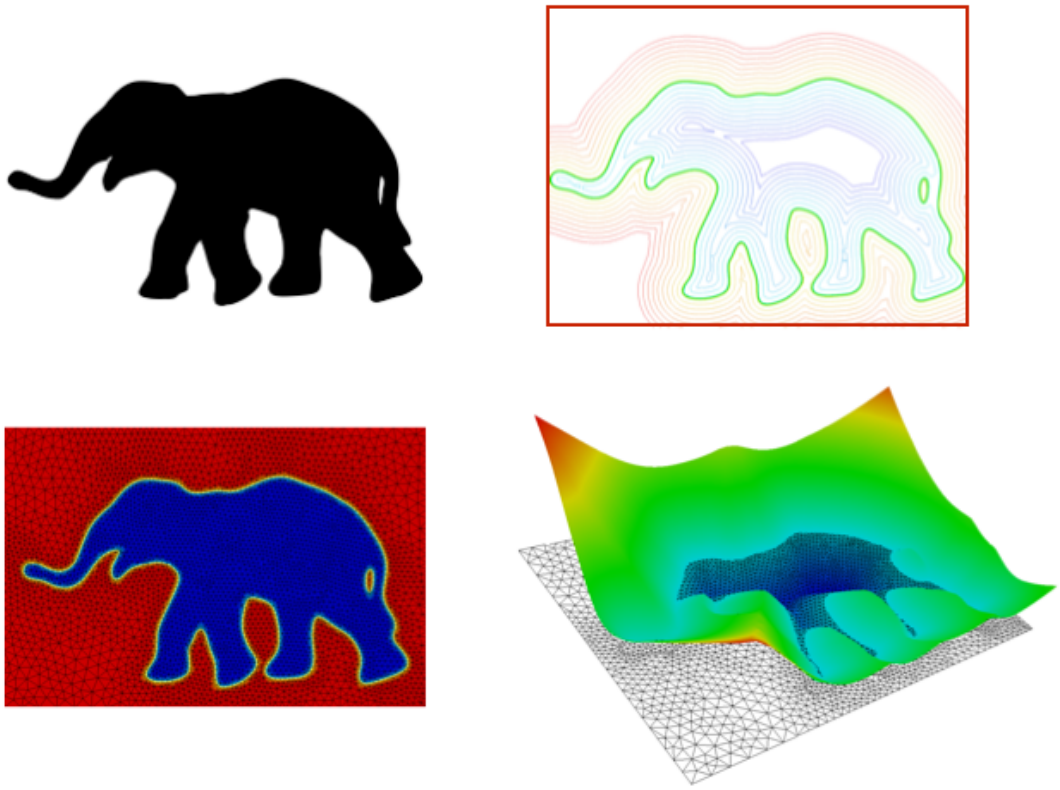


Figure 3: Top right: initial domain Ω in black color. Top left: isolines of the signed distance function to Ω. Bottom left: computation of the signed distance function to Ω on an unstructured mesh (the blue color stands for Ω). Bottom right: graph of the signed distance function to Ω.

any function $\phi : \mathbb{R}^d \to \mathbb{R}$ *verifying:*

$$\begin{cases} \phi(x) < 0 & \text{if } x \in \Omega, \\ \phi(x) = 0 & \text{if } x \in \partial\Omega, \\ \phi(x) > 0 & \text{if } x \in {}^c\overline{\Omega}. \end{cases}$$

The (Euclidean) *signed distance function* d_Ω to Ω is a level set function that gives the distance to the nearest point on $\partial\Omega$ at every point of the ambient space.

Definition 1.3.2. *The signed distance function* $d_\Omega(x) : \mathbb{R}^d \to \mathbb{R}$ *to* Ω *is defined as:*

$$\forall x \in \mathbb{R}^d,\ d_\Omega(x) = \begin{cases} -d(x, \partial\Omega) & \text{if } x \in \Omega, \\ 0 & \text{if } x \in \partial\Omega, \\ d(x, \partial\Omega) & \text{if } x \in {}^c\overline{\Omega}. \end{cases}$$

In the above formula, $d(x, \partial\Omega) = \inf_{y \in \partial\Omega} |x - y|$ denotes the usual Euclidean distance function to $\partial\Omega$. The signed distance function has been widely used in computational mathematics, for CAD modeling, surface reconstruction from scattered data [121], and last but not least for reinitializing level set algorithm [86].

Definition 1.3.3. *Let* $x \in \mathbb{R}^d$ *be a point and* $\Omega \subset \mathbb{R}^d$ *a bounded, Lipschitz domain. The set of projections* $\Pi_{\partial\Omega}(x)$ *of* x *onto* $\partial\Omega$ *is the set:*

$$\Pi_{\partial\Omega}(x) = \{z \in \partial\Omega, |z - x| = d(x, \partial\Omega)\}.$$

If $\Pi_{\partial\Omega}(x)$ *is a singleton, its unique element is called the projection of* x *onto* $\partial\Omega$ *and it is denoted by* $p_{\partial\Omega}(x)$*.*

The differentiability of the signed distance function is discussed below. The proof of the proposition can be found in [6].

Proposition 1.3.1. *Let* $x \in \mathbb{R}^d \setminus \partial\Omega$ *be a point. If* x *admits a unique projection* $p_{\partial\Omega}(x)$ *onto* $\partial\Omega$ *then the signed distance function* d_Ω *is differentiable at* x *and its gradient is given by:*

$$\nabla d_\Omega(x) = \frac{x - p_{\partial\Omega}(x)}{d_\Omega(x)}.$$

In particular, the signed distance function solves the following Eikonal equation:

$$|\nabla d_\Omega(x)| = 1,$$

at every point x *where it is differentiable. Moreover, if* Ω *is of class* $\mathcal{C}^1$ *then*

$$\nabla d_\Omega(x) = n(p_{\partial\Omega}(x)),$$

where $n(y)$ *stands for the unit normal at* y*, pointing outward to* Ω*.*

The signed distance function is thus a natural extension of the normal vector field. It can been proved that in the general case in which the signed distance function may

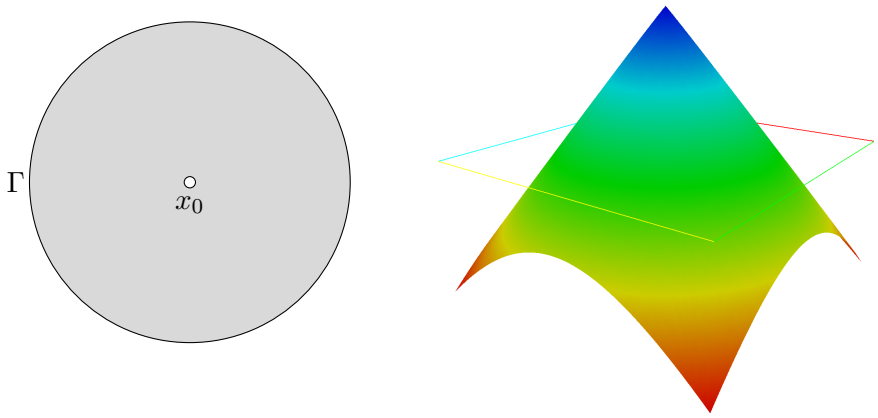


Figure 4: The signed distance function to a circle is not differentiable at the center x_0: each point on the boundary Γ belongs to the set $\Pi_{\partial\Omega}(x_0)$.

not be differentiable, it still solves the Eikonal equation in the sense of viscosity theory [29, 41].

1.3.1 Numerical computation of the signed distance function

The computation of the signed distance function is usually achieved by numerically solving an Eikonal type equation [94, 30]. This kind of approach has been widely developed in the context of level set methods. Fast and efficient approaches have been proposed in the Sethian's Fast Marching Method [95] and also the Fast Sweeping Method [122]. Another approach consist in turning the problem into a time dependent evolution process. In practice, the computation of the signed distance function is achieved by front propagation methods. The method used in this manuscript (introduced in [30]) is based on this point of view.

1.4 Definitions and useful notions about meshes

Mesh modeling is probably the most common way to account for shapes. For a complete overview about basic definitions and mesh generation techniques we refer to the monograph [44]. Here we limit ourselves to define the objects of interest: *simplicial meshes* in $\mathbb{R}^2, \mathbb{R}^3$ and *surface meshes* in $\mathbb{R}^3$. The definitions are taken from [30].

Definition 1.4.1. *Let $\Omega \subset \mathbb{R}^d (d = 2, 3)$ a bounded, open, polygonal domain. A simplicial mesh $\mathcal{T}$ of Ω is a finite collection $(K_i)_{i=1,\dots,N_\mathcal{T}}$ of closed d-simplices verifying the following requirements:*

1. *The elements of* $\mathcal{T}$ *cover* Ω*:* $\overline{\Omega} = \cup_{i=1}^{N_\mathcal{T}} K_i$*;*
2. *each simplex* K_i *of* $\mathcal{T}$ *has non empty interior:* $\mathring{K} \neq \emptyset$*;*
3. *if* $K_i, K_j \in \mathcal{T}$*,* $i \neq j$*, are distinct simplices then their interiors are disjoints:* $\mathring{K_i} \cap \mathring{K_j} = \emptyset$*;*
4. *if* $K_i, K_j \in \mathcal{T}$*,* $i \neq j$*, are distinct simplices then the intersection* $K_i \cap K_j$ *is: either a point or an edge shared by* K_i *and* K_j *(if* $d = 2$*), either a point or an edge shared by* K_i *and* K_j *or a triangle shared by* K_i *and* K_j *(if* $d = 3$*).*

The vertices and edges of the simplices K_i are called the vertices and the edges of $\mathcal{T}$. Obviously, a 2-simplex is a triangle, while a $3d$-simplex is a tetrahedra. The first condition assures that the discrete data $\mathcal{T}$ is a tessellation of the domain Ω, i.e. it is a good descriptor of its geometry; condition (2) is needed for avoiding degenerate elements; condition (3) avoids overlapping elements; condition (4) guarantees the conformity of the mesh. In the sequel we will call "invalid' a mesh not fulfilling one of the above listed properties.

Definition 1.4.2. *Let* $\Gamma \subset \mathbb{R}^3$ *be a compact polyhedral surface, with our without boundary. A surface mesh or triangulation* $\mathcal{S}$ *of* Γ *is a finite collection* $(\mathcal{S}_i)_{i=1,\ldots,N_\mathcal{S}}$ *of closed triangles* $T_i \subset \mathbb{R}^3$ *verifying the following properties:*

1. *The elements of* $\mathcal{S}$ *form a covering of* Γ*:* $\Gamma = \cup_{i=1}^{N_\mathcal{S}} T_i$*;*
2. *Each triangle* T_i *of* $\mathcal{S}$ *has non zero two-dimensional measure;*
3. *The intersection* $T_i \cap T_j$ *of every two distinct triangles* T_i*,*T_j *of* $\mathcal{S}$*,*$i \neq j$*, is the edge of a triangle of* $\mathcal{S}$*;*
4. *For every two distinct triangles* $T_i, T_j \in \mathcal{S}$*,*$i \neq j$*, the intersection* $T_i \cap T_j$ *is either a point, or a common edge of* T_i *and* T_j*.*

Generally speaking, meshes are generate for two main purposes: discrete representation of geometric data (as in computer graphics) and numerical computations using the mesh as support. In the first case, we will say that our mesh is of high 'quality' if it is a good approximation of the geometry of the underlying continuous object (so as for example if by refining the mesh geometric quantities defined on the mesh converge to their continuous counterpart). In this case we will refer to geometric meshes. In the second case, the 'quality' of the mesh will be judged in terms of the quality of the discrete solution of the numerical problem under investigation. For example, in Finite Elements computations, the quality of the mesh is related to a error estimation. In the case in which the mesh will be used as a support for a Finite Elements computation we will talk about computational meshes or Finite Elements meshes. In the course of this manuscript we will encounter several time Finite elements computations. A Finite element type computation relies on the following steps:

- Construction of a mesh as support of the computation;
- An interpolation step which constructs the 'finite elements' starting from the elements of the mesh;
- Construction of the linear system (mass matrix and right-hand side) corresponding to the discrete version of the continuous problem;
- Computation of the solution of the problem by solving the above assembled linear system.

In numerical computations one is often confronted with the problem of evaluating the discrepancy between two given triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$. Several notions of discrete distances are available in the literature. In the context of this work we will refer to the Hausdorff distance d_H, defined as the maximum:

$$d_H(\mathcal{T}_0, \mathcal{T}_1) := \max(\rho(\mathcal{T}_0, \mathcal{T}_1), \rho(\rho(\mathcal{T}_1, \mathcal{T}_0)). \tag{1.4.1}$$

Let p be a vertex of $\mathcal{T}_0$. The quantity $\rho(\cdot, \cdot)$ featured in (1.4.1) is defined as:

$$\rho(\mathcal{T}_0, \mathcal{T}_1) = \max_{p \in \mathcal{T}_0} d(p, \mathcal{T}_1).$$

In the above definition, $d(\cdot, \mathcal{T}_1)$ stands for the Euclidean distance function to the set of triangles of $\mathcal{T}_1$.

1.5 Linear Elasticity equations

Let us now introduce a particular mechanical system which will be sistematically employed in in the rest of the manuscript. Let $\Omega \subset \mathbb{R}^d (d = 2, 3)$ a bounded and Lipschitz domain fulfilled with an homogeneous isotropic elastic material. The set of points $x \in \overline{\Omega}$ describes the geometry of a material body in an unstressed configuration. Such a configuration is said to be a reference configuration for the body. Since Ω is instantaneously deformed under external loads, the deformation is described in terms of an application $\phi : \overline{\Omega} \to \mathbb{R}^d$ or equivalently in terms of the displacement field $u = \phi - Id$. The set of points $\phi(\overline{\Omega}) \subset \mathbb{R}^d$ represents the deformed configuration of the body under the deformation ϕ. The deformation is measured in terms of the Cauchy-Green strain tensor

$$C(\phi) = \nabla\phi^T \nabla\phi,$$

or equivalently in terms of the Green-Saint Venant tensor

$$E(\phi) = \frac{1}{2}(C - I), \text{ or equivalently } E(u) = \frac{1}{2}\left(\nabla u^T + \nabla u\right) + \frac{1}{2}\nabla u^T \nabla u.$$

The constitutive equation

$$\sigma(u) = D : E(u) \tag{1.5.1}$$

relating the internal stress $\sigma(u)$ and the deformation tensor $E(\phi)$ through the fourth order elasticity tensor D takes into account the material behavior. The classical theory of linear elasticity is based on two fundamental assumptions:

- in the range of small deformation the strain-displacement relation is described in terms of the linearized stress tensor $e(u) = \frac{1}{2}(\nabla u^T + \nabla u)$, which replaces $E(u)$;
- the constitutive material law is linear.

Under the assumption of linearity the relation (1.5.1) for an homogeneous isotropic elastic material drastically simplifies into the so called Hook law:

$$\sigma(u) = 2\mu e(u) + \lambda \mathrm{tr}(e(u))I,$$

depending only on two real coefficients λ, μ called the Lamé parameters.
The displacement field of a material body clamped at a part Γ_D of its boundary Γ and submitted to external load f on $\Gamma_N := \Gamma \setminus \Gamma_D$ is solution of the following boundary value problem:

$$\begin{cases} -\text{ div } (\sigma(u)) = f & \text{in} \quad \Omega, \\ \sigma(u) \cdot n = g & \text{on} \quad \Gamma_N, \\ u = 0 & \text{on} \quad \Gamma_D, \end{cases} \tag{1.5.2}$$

whose variational form reads:

$$\forall v \in H_0^1(\Omega)^d, \ \int_\Omega \sigma(u) : e(v)\, dx = \int_{\Gamma_N} gv\, ds + \int_{\Gamma_D} fv\, dx. \tag{1.5.3}$$

The following Korn inequality ensures the well-posedness of the problem (see [69] for the proof).

Theorem 1.5.1. *Let $\Omega \subset \mathbb{R}^d$ be a open Lipschitz domain. Let $v \in L^2(\Omega; \mathbb{R}^d)$ and $e(v) \in L^2(\Omega; \mathbb{M}_d)$. Then there exists a constant $L = L(\Omega)$ such that:*

$$||\nabla v||^2_{L^2(\Omega;\mathbb{M}_d)} \leq L(||v||^2_{L^2(\Omega;\mathbb{R}^d)} + ||e(v)||^2_{L^2(\Omega;\mathbb{M}_d)})$$

where $\mathbb{M}_d$ is the set of $d \times d$ matrices. Moreover, $v \in H^1(\Omega, \mathbb{R}^d)$.

The existence and uniqueness result (from [23]) is the following.

Theorem 1.5.2. *Let $\Omega \subset \mathbb{R}^d$ be an open Lipschitz domain. Let Γ_D be a measurable subset of $\Gamma = \partial\Omega$ of positive measure. Let $f \in L^{\frac{6}{5}}(\Omega)$ and $g \in L^{\frac{4}{3}}(\Gamma \setminus \Gamma_D)$. Then the problem (1.5.3) admits a unique solution u belonging to the space $H_0^1(\Omega)^d = \{v \in H^1(\Omega); v = 0$ a.e. on $\Gamma_D\}$.*

Since the equations under considerations are elliptic, additional regularity on the data (the domain Ω and the function f, g) leads to additional regularity of the solution. More precisely for the pure displacement problem (i.e. when $g = 0$) we have the following statement (see [23]).

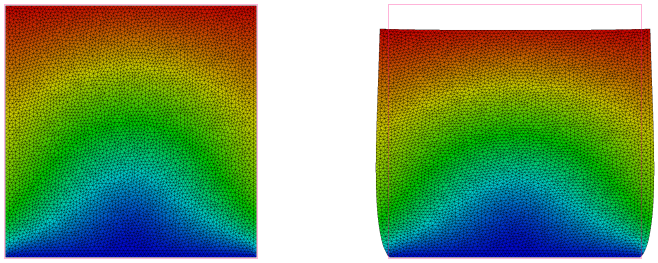


Figure 5: Square domain before deformation (left) and in deformed configuration (right).

Theorem 1.5.3. *Let $\Omega \subset \mathbb{R}^d$ be an open Lipschitz domain. Let $p \geq \frac{6}{5}$. If Γ is of class C^{m+2} and $f \in W^{m,p}$ then the solution u of* (1.5.3) *belongs to the space $W^{m+2,p}(\Omega)$.*

The above regularity result does not apply to the general problem with mixed boundary condition when a transition between boundary conditions of different nature occurs. Otherwise the theorem is still valid if the displacement and the traction boundaries do not intersect.

Chapter 2

An optimization method for elastic shape matching

Contents

2.1 Introduction

In this chapter we investigate the problem of $2d$ and $3d$ shape matching. This chapter contains the first original contribution of the manuscript. This is a joint work with Maya de Buhan, Charles Dapogny and Pascal Frey.

Shape morphing or matching arises in a wide variety of situations in areas from biomedical engineering to computer graphics, scientific computing, robotics, pattern

recognition. Beyond the specific task of each particular application, the general issue is to find one transformation from a given 'template' shape Ω_0 into a 'target' shape Ω_T. Crucial components of a shape matching procedure are the way for accounting for shapes and the development of the mathematical model which drives the deformation. Such a transformation may be used as a means to appraise how much Ω_0 and Ω_T differ from one another -for instance in shape retrieval, classification, recognition or tracking- or to achieve physically the transformation from Ω_0 to Ω_T (in shape registration, reconstruction, or simplification). See for instance [111] and references therein for an overview of several related applications. Understandably enough, a great deal of work has been devoted to shape matching, and many methods have been proposed for addressing this problem; we limit ourselves to mentioning a few approaches. In [12] the authors start by distributing sample points on the contour of both shapes, that will be matched according to their 'shape context', namely a shape descriptor defined as an histogram of the relative position of the sampled points with respect to a given point on the shape. They eventually infer a global transformation from this point-to-point correspondence. However, note that reducing curves or surfaces to sample points risk to discard higher order geometric informations (tangent plane, curvature), leading to uncorrect results.

In the field of computational anatomy, a series of articles (see e.g. [10, 40, 54]) have suggested to describe the deformation of Ω_0 onto Ω_T via a diffeomorphic map. The restraint to diffeomorphims prevents discontinuous or self intersecting deformations. This approach leads to the so-called Large Deformation Diffeomorphic Metric Mapping (LDDMM) method which has received a lot of interest in the last years, especially for dealing with medical data. The LDDMM approach considers shapes embedded in a metric space and describes the diffeomorphism between two items Ω_0 and Ω_T as the dynamic flow of a velocity field v. The search for v is then casted as an optimal control problem. The criterion to be minimized is the sum of a matching functional -representing the cost of the flow which transports Ω_0 onto Ω_T- and of a regularization term, guaranteeing the smoothness of the deformation. The structure of metric space allows the definition of similarity measures between shapes, in terms of geodesics in the shape space. The LDDMM formulation produces a global deformation of the ambient space, which in particular matches the shapes of interest into each other. In [83] the energy functional associated to a LDDMM problem is used for defining a measure of similarity between two shapes, defined as the geodesic path in the space of diffeomorphism. The LDDMM framework has been used for landmarks-based matching using the spline interpolation theory [65], unlabelled point-sets [51], $3d$ stacks of density images [11, 10], curves [52, 50] or surfaces [49, 108]. In [108] a theoretical framework is provided for encoding surfaces embedded in $\mathbb{R}^3$ as currents; the optimal control problem is defined consequently as matching of currents. The resulting mapping is used to study features of organs, detect anomalies, etc.

Several shape matching methods use landmarks or anchor points for which the correspondence is supposed known; then the deformation is guided by this a priori information. Depending on the application, these feature points are defined manually (for example in the case of anatomical landmarks, see Chapters 5 and 6) or by

automatic extraction, for instance as critical points of a well-suited scalar function defined on the shape. However the automatic detection of feature points and the construction of a meaningful scalar function are not general and depend on the structure of the shape at stake.

Conformal geometry theory have been used for translating a $3d$ shape matching problem into a $2d$ one by conformal parametrization of the boundary, via harmonic maps, conformal maps, least square conformal maps [113, 56, 73]. In a series of articles devoted to shape matching and retrieval, $3d$ shape matching is achieved by extraction and comparison of topological features, namely Reeb graphs [61], or others topological indicators. However, a Reeb graph is computed from the analysis of the isosurfaces of a real-valued function defined in a manifold; for many shapes the extraction of such a function is not a trivial task.

More recently, in the field of Computer Graphics, the optimal transport point of view has been used to displace an input tetrahedral mesh onto a given object [70, 99] and to register $2d$ images [58]. In the optimal transport point of view a measure (derived from a density function supported by a computational mesh or defined as a sum of Dirac masses) is displaced into another by minimizing the celebrated Monge-Kantorovitch optimal mass transportation cost. Methods from optimal transport have the merit to be pose independent, which is of tremendous importance for searching shape similarity, and to allow topology changes.

Our method has much in common with [97, 96], proposing a physically inspired deformation model which is supported by a computational mesh. For the sake of generality, the proposed method does not make use of any any preprocessing feature extraction. For preventing merging and splitting during the deformation, the proposed framework does not allow topology changes. Our problem is stated as follows: given a 'template' shape Ω_0, numerically described by means of a (conforming) computational mesh, and a 'target' shape Ω_T, known only via the signed distance function to its boundary, we aim at deforming (iteratively) the mesh of Ω_0 into a computational mesh of Ω_T. Such a technique could be applied, for instance, to the reconstruction of a computational mesh Ω_T from invalid data, to transport quantities of interest from Ω_0 to Ω_T, etc. To achieve our purpose, we rely on a method which has much in common with that of [7], borrowing techniques from shape optimization, and more generally optimal control. Under the assumption that Ω_0 and Ω_T share the same topology, the desired transformation from Ω_0 to Ω_T is realized as a sequence of elastic displacements, which are obtained by minimizing an energy functional based on the distance between Ω_0 and Ω_T. In doing so, it is expected that the deformation will be easier to achieve in numerical practice, and in particular by limiting the troubles due to mesh tangling.

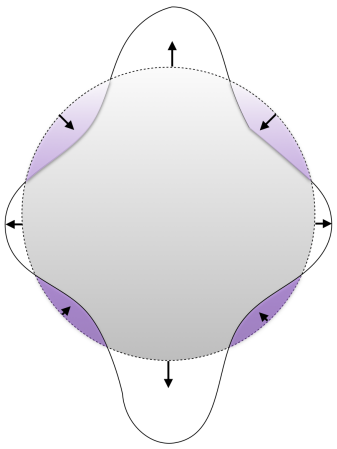


Figure 1: Deformation of the template shape decreasing the value of J.

2.2 Shape matching without constraint on the displacement

Let $\Omega_0, \Omega_T \subset \mathbb{R}^d (d = 2, 3)$ be respectively 'template' and 'target' shapes, i.e. bounded Lipschitz domains. We assume that they share the same topology but they are not necessarely close from one other. Our purpose is to map Ω_0 onto Ω_T, which we achieve borrowing techniques from shape optimization, and in general from optimal control theory. Let us present the theoretical framework of the method. The discrepancy between a reference shape Ω and the target shape Ω_T is measured by the following functional $J(\Omega)$ of the domain:

$$J(\Omega) = \int_\Omega d_{\Omega_T}(x)dx, \tag{2.2.1}$$

which involves the signed distance function d_{Ω_T} to Ω_T, defined in Section 1.3.

Note that this functional is defined on the whole domain of interest Ω. In order to decrease the value of $J(\Omega)$, the domain Ω must expand in the regions of the ambient space $\mathbb{R}^d$ where d_{Ω_T} is negative (that is, in the regions comprised in Ω_T), and to retract in those where it is positive. By deforming the template shape as above we can always obtain better values for J until the deformed template shape reaches the target shape Ω_T. Therefore we get the following intuitive existence result.

Theorem 2.2.1. *The functional $J(\Omega)$ has a unique, global minimizer $\Omega = \Omega_T$ over the space $\mathcal{O}_{ad} := \{\Omega \subset \mathbb{R}^d, \Omega \text{ open }\}$.*

Proof. Let Ω^* be an optimum for J. From Proposition 1.2.1 and Theorem 1.2.1, the first order optimality condition for the functional J leads to the equality:

$$d_{\Omega_T}(x) = 0, \text{ for all } x \in \partial\Omega^*.$$

Therefore $\Omega^* = \Omega_T$. □

Note that the same result holds when the signed distance function is replaced in 2.2.1 with any level set function of the domain Ω_T. It is then expected that an iterative (e.g. gradient-based) algorithm devoted to the minimization of $J(\Omega)$, starting from Ω_0, will lead to an interesting way to transform Ω_0 into Ω_T. In next section we will describe the general framework of the iterative procedure.

2.3 General procedure for the minimization of J

From the expression of the shape derivative (see Section 1.2.1), the vector field

$$\theta = -d_{\Omega_T}(x)n \tag{2.3.1}$$

is a natural descent direction for J. Indeed, by Theorem 1.2.1, it guaranties that for $\varepsilon > 0$ small enough:

$$J((I + \varepsilon\theta)\Omega)) = J(\Omega) - \varepsilon \int_{\partial\Omega} d^2_{\Omega_T}(s)\, ds + o(\varepsilon) < J(\Omega).$$

This paves the way for an iterative algorithm, producing a sequence $(\Omega_k)_{k=0,\dots}$ of shapes, which are 'closer and closer' to Ω_T: at each step, Ω_k is updated according to

$$\Omega_{k+1} = (I + \theta_k)(\Omega_k), \text{ where } \theta_k \text{ is (an extension to } \Omega_k \text{ of) } d_{\Omega_T} n_{\Omega_k}, \tag{2.3.2}$$

where n_{Ω_k} stands for the unit normal vector to $\partial\Omega_k$, pointing outward Ω_k. From the discrete point of view, we aim at generating a sequence of meshes $(\mathcal{T}_k)_{k=0,\dots}$ which boundaries $(\mathcal{S}_k)_{k=0,\dots}$ are closer and closer to the 0-level set of the scalar function d_{Ω_T}. At each iteration the mesh $\mathcal{T}_k$ is updated by pushing its vertices according to the descent direction:

$$\forall x \text{ vertex of } \mathcal{T}_k,\ x \mapsto x + \theta_k(x). \tag{2.3.3}$$

The formal procedure described above boils down in deforming a shape Ω in the negative direction of the $L^2(\partial\Omega)^d$ gradient of the differential $\theta \mapsto J'(\Omega)(\theta)$:

$$J'(\Omega)(\theta) = \langle f, \theta \cdot n \rangle_{L^2(\partial\Omega)^d}. \tag{2.3.4}$$

Unfortunately, the descent direction associated to the vector field (2.3.1) reveals unsuited for at least two reasons:

- the vector field featured in (2.3.1) is defined only on the boundary of Ω; it has therefore to be extended to Ω as a whole so that it can be a guide for displacing the vertices of $\mathcal{T}$;
- if no particular attention is paid to this extension, the extended displacement field may impose an important stretching in Ω, making the motion of the vertices

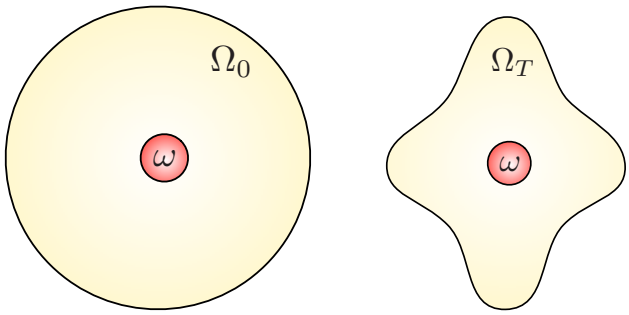


Figure 2: Target and template shapes sharing a common fixed subset ω.

of $\mathcal{T}$ via (2.3.3) impossible to achieve without invalidating the mesh.

For overcoming both this difficulties we have investigated two alternative -but close in essence- strategies. The first one is based on an extension and regularization of the descent direction (2.3.1) by using the gradient of $\theta \mapsto J'(\Omega)(\theta)$ associated to another inner product (Section 2.4). The latter consists in restraining the space of admissible deformations of the template shape Ω_0 to elastic displacements generated by suitable surface loads acting on its boundary (Section 2.5). Both methods make use of the operator of linear elasticity which we briefly recalled in Section 1.5.

2.4 Shape matching with extension and regularization of the descent direction

Let $\Omega_0, \Omega_T \subset \mathbb{R}^d$, be respectively 'template' and 'target' shapes. In the present context we assume that they share the same topology but they are not necessarily close from another. The key idea is to use the gradient of the mapping $\theta \mapsto J'(\Omega)(\theta)$ associated to another inner product instead of the one featured in (2.3.4). This *velocity extension - regularization* issue is quite classical in shape optimization (see [35] and references therein), and can be thought of as an efficient preconditioning of the naive procedure (2.3.2). The general strategy is described in the following. Suppose that the space of admissible deformations is endowed with a structure of Hilbert space V. Let $\langle \cdot, \cdot \rangle_V$ be the inner scalar product in V. Therefore, we can consider the following linear form $l(\cdot)$ acting on elements of V:

$$\forall v \in V,\ \ l(v) = J'(\Omega)(v).$$

Now let $a_V(\cdot, \cdot)$ a suitable symmetric, V-elliptic bilinear form acting on elements of V. The Lax-Mailgram theorem assures the existence of a unique $u \in V$ solution of the variational problem:

$$\forall v \in V,\ \ a_V(u, v) = f(v).$$

The vector field $-u$ is naturally a descent direction for J since:

$$J'(\Omega)(-u) = -a_V(u,u) < 0.$$

A natural choice is the space $V = H^1(\Omega)^d$ equipped with the inner product:

$$\langle v, w \rangle_V = \int_\Omega a\nabla u \cdot \nabla v + vw, a \in \mathbb{R}, a > 0.$$

Another suitable choice in our context is to use the extension inherited from the operator of liner elasticity introduced in Section 1.5. Therefore, assume that all the considered shapes Ω are filled with a linear elastic material. Also, imagine that any such shape Ω contains a given subset $\omega \Subset \Omega$ on which it is clamped. We now obtain a descent direction for $J(\Omega_k)$ as the unique solution u_Ω belonging to $H^1_\omega(\Omega)^d := \{v \in H^1(\Omega)^d, v = 0 \text{ in } \omega\}$ of the linearized elasticity system:

$$\forall v \in H^1_\omega(\Omega)^d, \quad \int_\Omega \sigma(u_\Omega) : e(v)\, dx = -J'(\Omega)(v) = -\int_{\partial\Omega} d_{\Omega_T} v \cdot n\, ds. \tag{2.4.1}$$

This vector field u_Ω is naturally a descent direction for $J(\Omega)$ since $J'(\Omega)(u_\Omega) \leq 0$ thanks to (2.4.1), and its advantages over the 'natural' deformation field θ defined in (2.3.2) are twofold:

1. u_Ω is defined on the whole shape Ω; owing to the regularizing effect of elliptic equations, it is intrinsically smoother than $\theta = -d_{\Omega_T} n$ (see for instance [23]);

2. Owing to the mechanical features of elastic displacements (notably their 'rigidity'), it is expected that u_Ω will be more amenable to the displacement of the mesh $\mathcal{T}$ into a valid mesh via (2.3.3); see e.g. [8] for an example of use of elastic displacements in the context of mesh displacement.

Remark 2.4.1. *From the numerical point of view, the choice of a subset ω corresponds to a global alignment of shapes (cf. Figure 2). This restriction we used to guarantee the well-posedness of Problem (2.4.1) could be replaced by adding a 0^{th}-order term.*

Remark 2.4.2. *Note that by construction the boundaries $\partial\omega$ and $\partial\Omega$ do not intersect, preventing the loss of regularity at the interface between boundary conditions of different nature.*

2.4.1 Numerical issues

As far as the numerical setting is concerned, the template shape Ω_0 is discretized as a simplicial mesh (i.e. a triangulation filled with tetrahedra), and the target shape Ω_T is supplied through its signed distance function, e.g. as a $\mathbb{P}^1$ piecewise affine function on the fixed mesh $\mathcal{T}_D$ of a large computational domain D.

Starting from the template shape Ω_0 we perform a standard gradient descent algorithm with adaptive step size in order to get a sequence of pairs $(\Omega_k, \mathcal{T}_k)$ of domains and their corresponding meshes with decreasing values of $J(\Omega_k)$. The algorithm stops when the step size is smaller than a fixed tolerance ε. The general scheme is summarized as follows:

Start from an initial shape Ω_0,
for $k = 0, \cdots$ **convergence**

1. compute the state u_{Ω_k} of the considered PDE system on Ω_k by FEM discretization;
2. compute the shape gradient $J'(\Omega_k)$ and infer a descent direction θ_k for the shape functional;
3. perform a line search for getting the optimal step size τ_k;
4. advect the shape Ω_k according to this displacement field, so as to get

$$\Omega_{k+1} = (I + \tau_k \theta_k)(\Omega_k).$$

Remark 2.4.3. *The global mapping u_Ω from Ω_0 to Ω_T is easily recovered by the composition of the different displacements $(u_k)_{k=0,\dots}$ between each iteration.*

Remark 2.4.4. *The only information required about the target shape Ω_T is the datum of its signed distance function which can be defined on a possibly non-conforming mesh (e.g. showing small gaps, overlapping entities, etc.).*

Remark 2.4.5. *The computational meshes used to perform the calculation are non uniform; they are refined in the vicinity of the boundaries according to a curvature based sizing function and coarsened in the interior of the domain. This has proven to prevent severe distorsion/tangling of the elements (avoiding the need to remesh the domain) and hence to increase the efficiency of the overall algorithm.*

Note that by updating the mesh at each iteration we are able to devise a deformation between two shapes which are not close from another by using a linear elastic model, which accounts in principle for 'small' displacements. Let us now present several numerical examples to show the performances of the method.

2.4.2 Numerical examples

In all the proposed examples, the calculation of the signed distance function to Ω_T is performed using the algorithm [33]. At first, we consider the $2d$ test case depicted in Figure 3. Both target and template meshes are embedded in a unit computational box with dimensions $[0, 1]^2$. The set ω chosen for aligning Ω_0 and Ω_T is a small disk located in the interior of both shapes.

The template mesh $\mathcal{T}_0$ has about 1 200 edges, and the convergence of the gradient descent procedure is obtained in 1 500 iterations for a tolerance $\varepsilon = 1.e^{-6}$.

Note that the algorithm is able to recover isolines of high curvature resulting in an excellent matching, even for template and target shape very far from another. Also, the use of linear elasticity equations makes the deformation smooth preserving the quality of the mesh under deformation.

The quantitative analysis of the error is performed by computing the discrepancy between the boundary Γ of deformed template Ω_k and the target shape $\partial\Omega_T$. The discrepancy $\mathcal{D}$ between Γ and $\partial\Omega_T$ is evaluated by the following L^2 error:

$$\mathcal{D}(\Gamma, \partial\Omega_T) = \left(\frac{1}{|\Gamma|} \int_\Gamma d^2(s, \partial\Omega_T)\, ds \right)^{\frac{1}{2}}, \tag{2.4.2}$$

where $|\Gamma|$ is the measure of Γ and $d(\cdot, \partial\Omega_T)$ is the Euclidean distance to $\partial\Omega_T$.

The discrepancy $\mathcal{D}(\partial\Omega_k, \partial\Omega_T)$ calculated on the boundary of the resulting shape Ω_{1500} equals $5.7e^{-4}$ (much smaller than the minimal mesh size), revealing an excellent recovering of Ω_T.

Figure 4 depicts the morphing between the same template mesh onto several silhouettes. Figure 5 show the contour superposition between the deformed template mesh and the contour of the target shape model under consideration. The convergence analysis is detailed in Table 2.1. The convergence history is reported in Figure 6.

Next, we consider a $3d$ example; see Figure 7. Both the target and the template meshes are embedded in a unit computational box $D = [0, 1]^3$. The shapes Ω_0 and Ω_T are aligned by choosing a small ball ω in $\Omega_0 \cap \Omega_T$ as for the subset ω. The template mesh $\mathcal{T}_0$ has about 9 000 triangles, and 1 300 iterations of the gradient descent algorithm have been performed to achieve convergence for a tolerance $\varepsilon = 1.e^{-6}$, running in a few minutes on a standard laptop computer. The L^2 discrepancy $\mathcal{D}(\partial\Omega_{1300}, \partial\Omega_T)$ calculated on the boundary of the final shape $\Omega_{1\,300}$ equals $5.04e^{-4}$ (again, much smaller than the minimal mesh size).

In Figure 8 a spherical template mesh is mapped onto the *Igea* model. The convergence analysis is detailed in table 2.1. The convergence history is reported in Figure 10.

In Figure 9 an ellipsoid template mesh is mapped onto a shape of hippocampus obtained from segmentation of medical images. The convergence history is reported in Figure 10. The convergence analysis relative to all the considered examples is detailed in table 2.1. We refer to Chapter 7 for others examples using medical data.

Remark 2.4.6. *We own from Proposition 1.3.1 that the signed distance function to Ω_T is not differentiable at points of the space having multiple projections over $\partial\Omega_T$, making troubling the definition of the shape derivative. This lack of differentiability does not create ambiguities here. Indeed if a point x admits multiple projection points realizing the same distance to the boundary $\partial\Omega_T$, the expression of the shape derivative*

to this point will depend only on the values of the signed distance function and not on the selected projection point.

Test case $\cdots$	$L2$ error	Elements	Iterations
1. Figure 3	$5.7e^{-4}$	1 200	1 500
2. Figure 4 (a)-(c)	$1.2e^{-03}$	1 200	750
3. Figure 4 (d)-(f)	$1.0e^{-03}$	1 200	500
4. Figure 4 (g)-(i)	$1.2e^{-03}$	1 200	2 300
5. Figure 7	$5.0e^{-4}$	9 000	1 300
6. Figure 8	$2.3e^{-4}$	117 000	1 500
7. Figure 9	$7.9e^{-3}$	36 000	1 700

Table 2.1: Convergence analysis for the proposed test cases. Second row: error computation using (2.4.2). Third row: number of elements (triangles in $3d$, edges in $2d$) of the template mesh. Fourth row: number of iterations needed to reach convergence.

2.5 Shape matching with constraint on the displacement

Let us now describe an alternative, perhaps more intuitive, strategy for devising the minimization of the functional 2.2.1. The procedure described in the previous section accounts for deforming a shape by rectracting/expanding its boundary by reacting to surface loads imposed on its boundary. At each iteration of the procedure, an elastic descent direction is computed by submitting the shape to surface loads which magnitude is proportional to the value of the signed distance function in the vicinity of the considered point. It seems then natural to try to accomplish this procedure by restraining the space of admissible deformations to those which are obtained as elastic displacements generated by surface loads acting on the boundary. This point of view leads to a minimization problem which is very similar to the one defined in the previous section, except that it is casted over a space $\mathcal{G}_{ad}$ of admissible loads and the deformations are parametrized as those which are generated by loads in $\mathcal{G}_{ad}$. Indeed, assume again that all the considered shapes Ω are filled with a linear elastic material and that any such shape Ω is clamped on a part $\Gamma_D \subset \partial\Omega$ of its boundary. In this context, the clamped part Γ_D plays the same role then the subset ω of the previous section. Let also $U_1 \Subset U_2 \subset \mathbb{R}^d$ be two open neighborhoods of Γ_D, and $\chi : \mathbb{R}^d \to \mathbb{R}^d$ be a smooth function enjoying the properties:

$$\chi \equiv 0 \text{ on } U_1, \;\; 0 \leq \chi \leq 1, \;\; \chi \equiv 1 \text{ on } {}^c\overline{U_2}.$$

The functional under consideration is then:

$$J(g) = \int_{(I+\chi u_{\Omega,g})(\Omega)} d_{\Omega_T}\, dx, \tag{2.5.1}$$

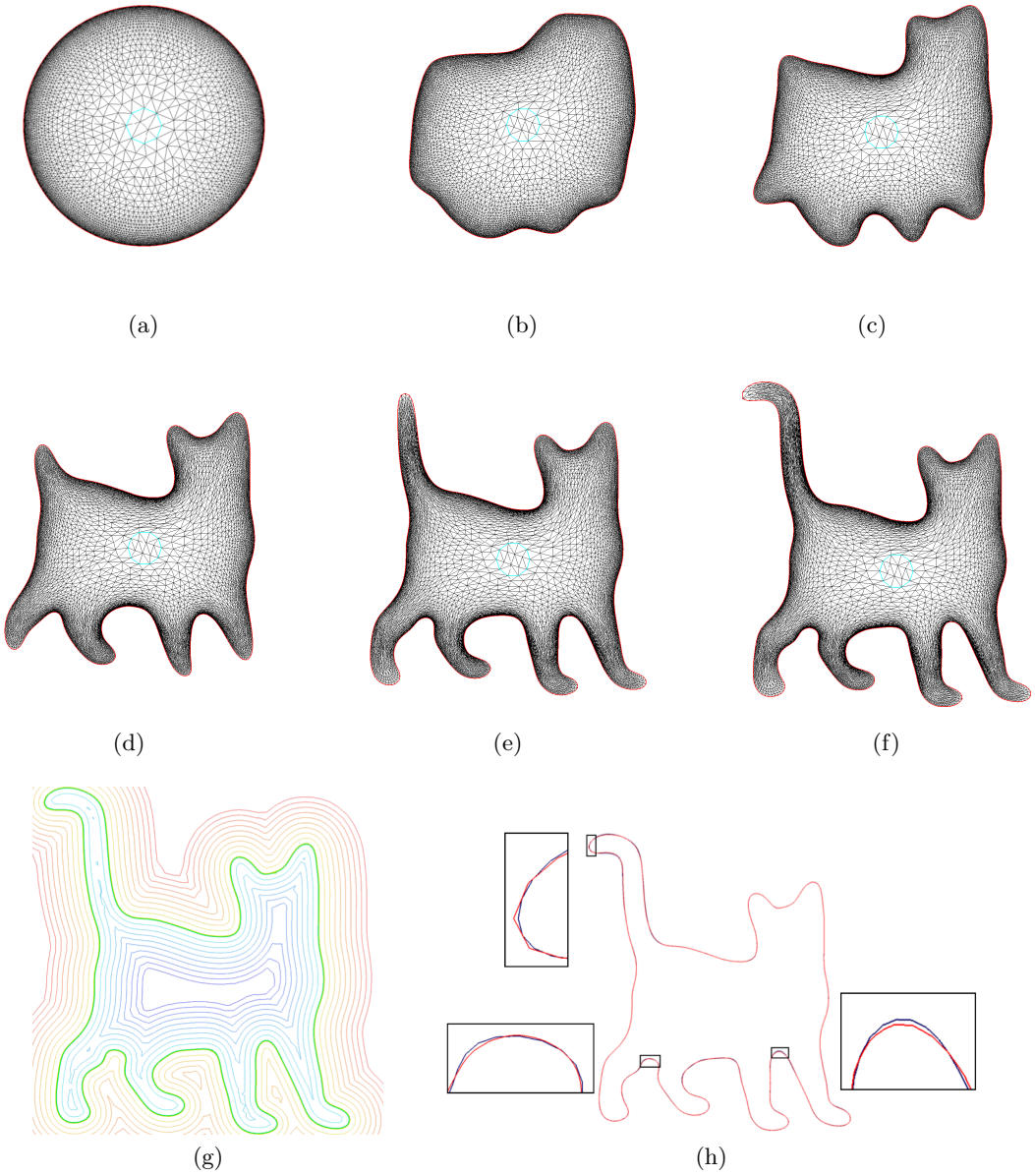


Figure 3: An example in 2D: (a) Template shape Ω_0. Deformed shape Ω_k for (b) $k = 40$; (c) $k = 60$; (d) $k = 200$; (e)$k = 300$; (f) $k = 1500$; (g) Isovalues of the signed distance function to the target shape Ω_T defined on the fixed mesh $\mathcal{T}_D$. (h) Discrepancy between Ω_T and Ω_{1500}.

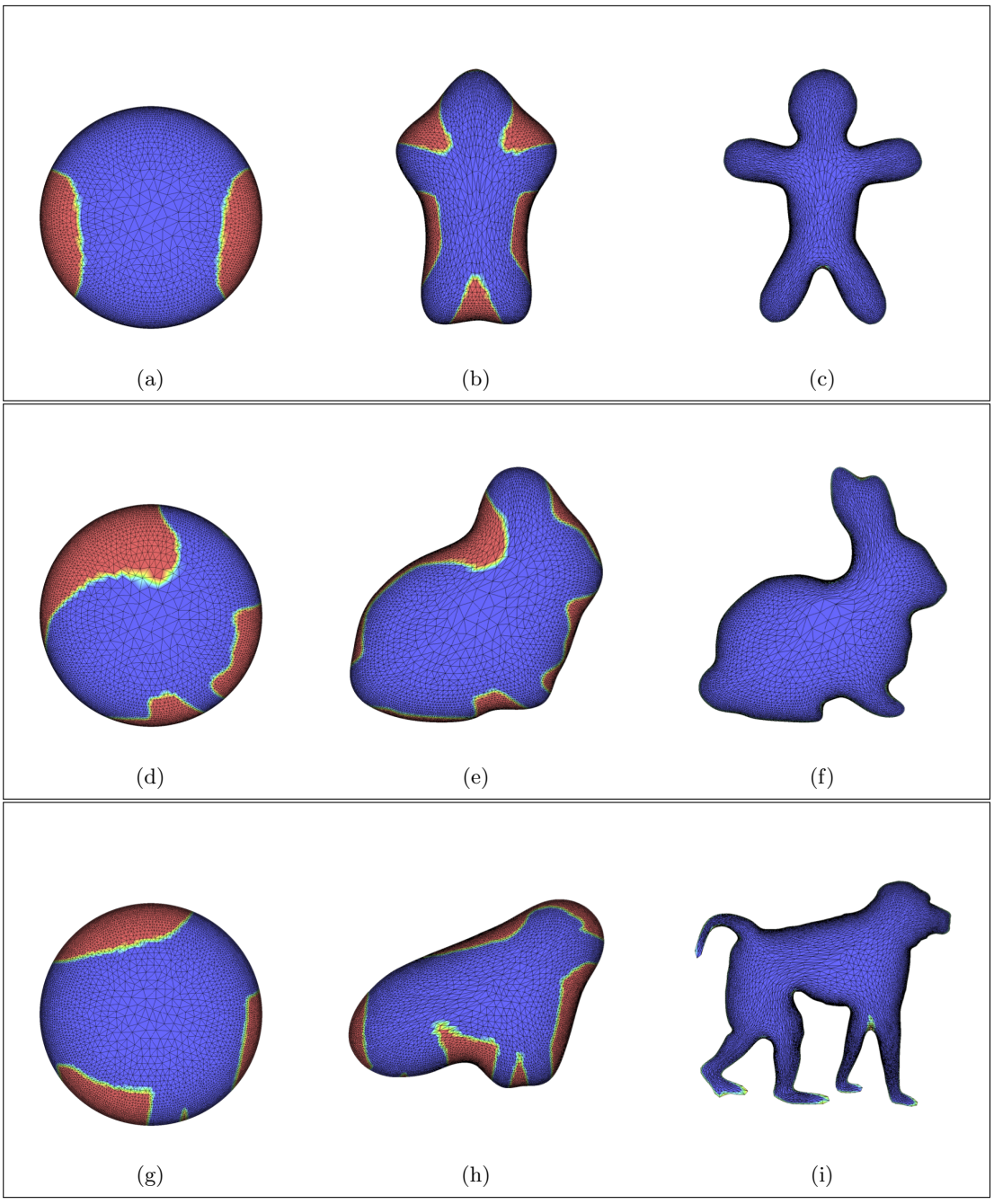


Figure 4: Three examples in 2D: (a)-(d)-(g) Template shape Ω_0 and discrepancy with the target shape. Deformed shape Ω_k for (b) $k = 90$; (e) $k = 60$; (h) $k = 200$. Deformed shape Ω_k for (c) $k = 750$; (f) $k = 500$; (i) $k = 2\,300$.

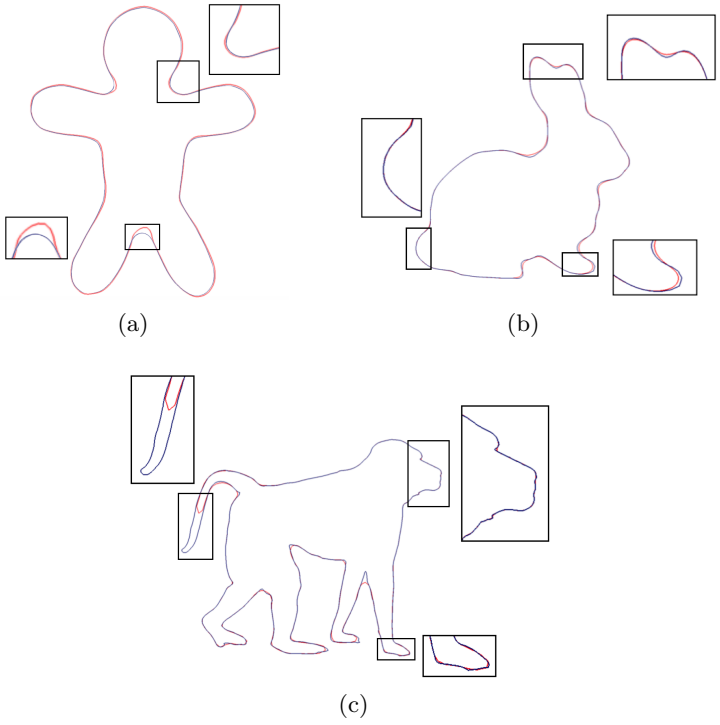
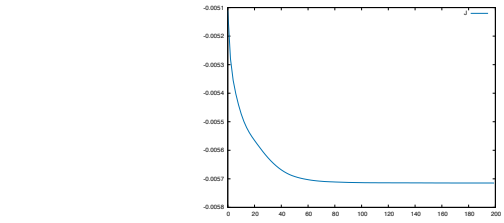
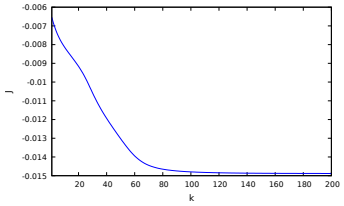


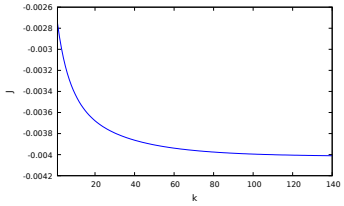
Figure 5: Discrepancy between Ω_T and the deformed template shape for the test cases of Figure 4.



(a)



(b)



(c)

Figure 6: Convergence history for the $2d$ test cases: objective functional $J(\Omega_k)$ versus number of iterations k. (a) Figure 3. (b) Figure 4-top row. (c) Figure 4-middle row.

where g belongs to a set $\mathcal{G}_{ad} \subset H^{-1/2}(\partial\Omega)$ of sufficiently regular loads and $u_{\Omega,g}$ is the solution belonging to $H_0^1(\Omega)^d := \{v \in H^1(\Omega)^d, v = 0 \text{ on } \Gamma_D\}$ of the variational problem:

$$\forall v \in H_0^1(\Omega)^d, \quad \int_\Omega \sigma(u_\Omega) : e(v)\, dx = -\int_\Gamma gv\, ds. \tag{2.5.2}$$

Note that J is very similar to the functional of section, except that we impose the displacement vector field to be generated by surface loads g applied on Γ. In the narrow sense, the functional 2.5.1 is not a functional of the domain. However, starting from a reference shape Ω we can put into correspondence a surface load g and the deformed domain $\Omega_{u_g} := (I + u_g)(\Omega)$ obtained by deforming Ω via the elastic displacement induced by g. In this sense, casting the minimization problem over $\mathcal{G}_{ad}$ is equivalent to cast it on the set of admissible shapes:

$$\mathcal{O}_{ad} := \{\Omega_g \in \mathbb{R}^d, \exists g \in \mathcal{G}_{ad} \text{ such that } \Omega_g = (I + u_g)(\Omega)\}.$$

We can therefore borrow the shape optimization techniques described in Chapter 1 for achieving the minimization of g.

Remark 2.5.1. *For the expression $(I + \chi u_{\Omega,g})(\Omega)$ to even make sense, we need $\chi u_{\Omega,g}$ to enjoy (at least) $W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)$ regularity. From a theoretical point of view, upon higher regularity assumptions on Ω and g for it is well-known (see Section 1.5 and reference therein) that the solution to the linear elasticity problem (1.5.2) is 'smooth' far from the areas where a transition between boundary conditions of different natures occur, and χ vanishes on those areas.*

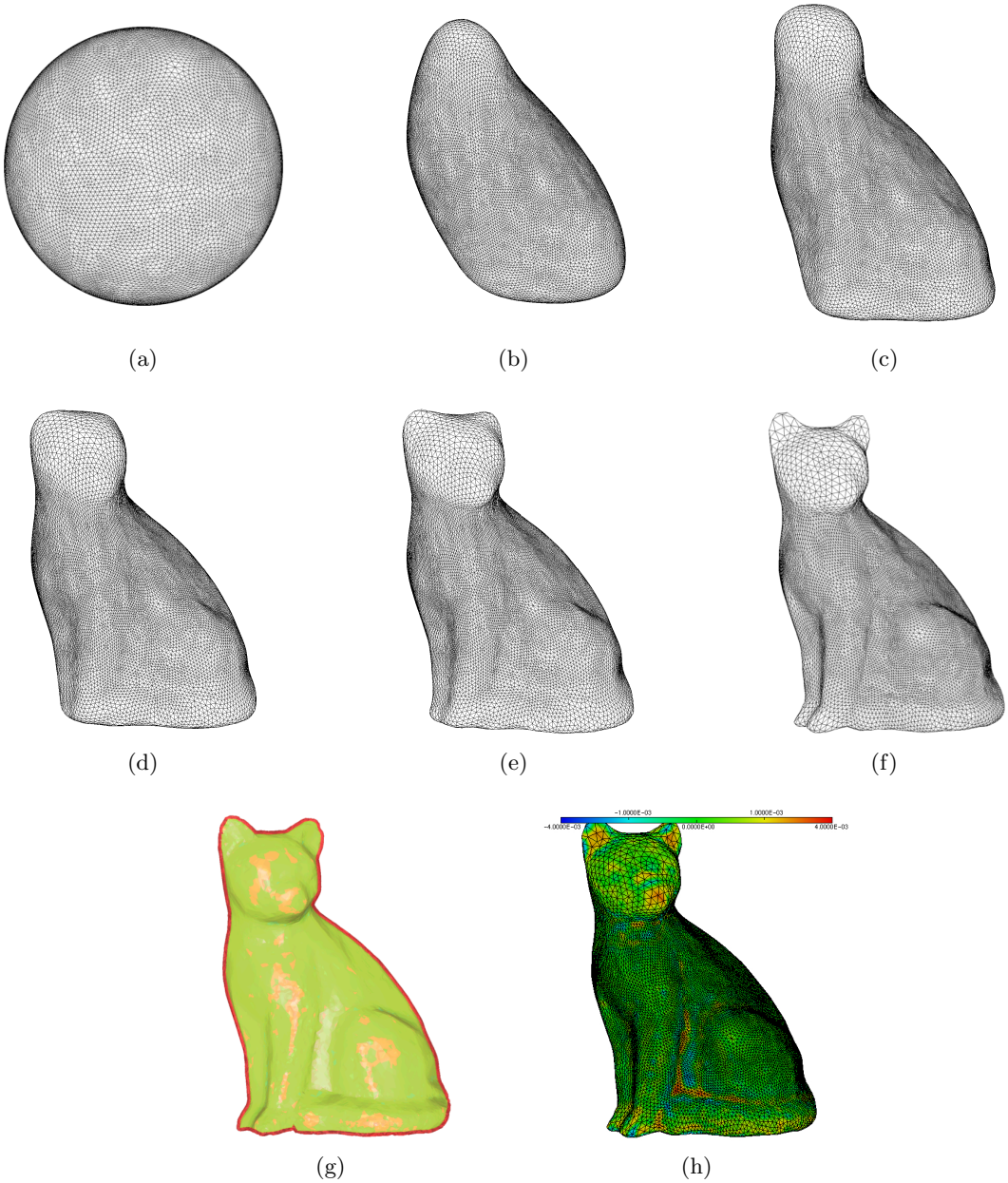


Figure 7: An example in 3D: (a) Template shape Ω_0. (b)-(f) Evolution of the template shape. (g) Target shape Ω_T as the zero level set of the signed distance function. (h) Discrepancy between Ω_T and Ω_{1300}.

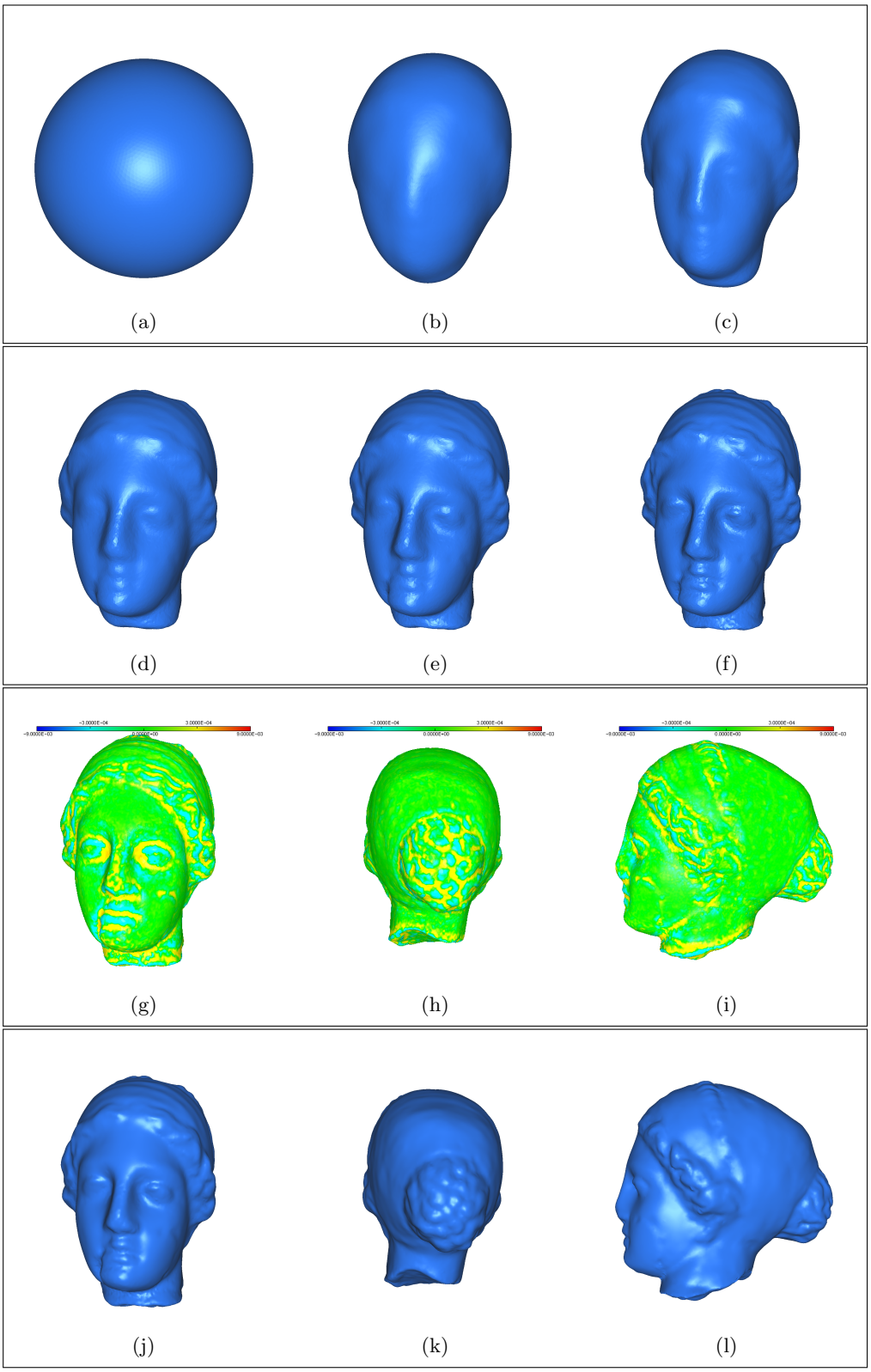


Figure 8: An example in 3D: (a) Template shape Ω_0. (b)-(e) Evolution of the template shape. (f) Deformed shape Ω_k for $k = 1\,500$. (g)-(i) Discrepancy between Ω_T and Ω_{1500}.(j)-(l) Target shape Ω_T.

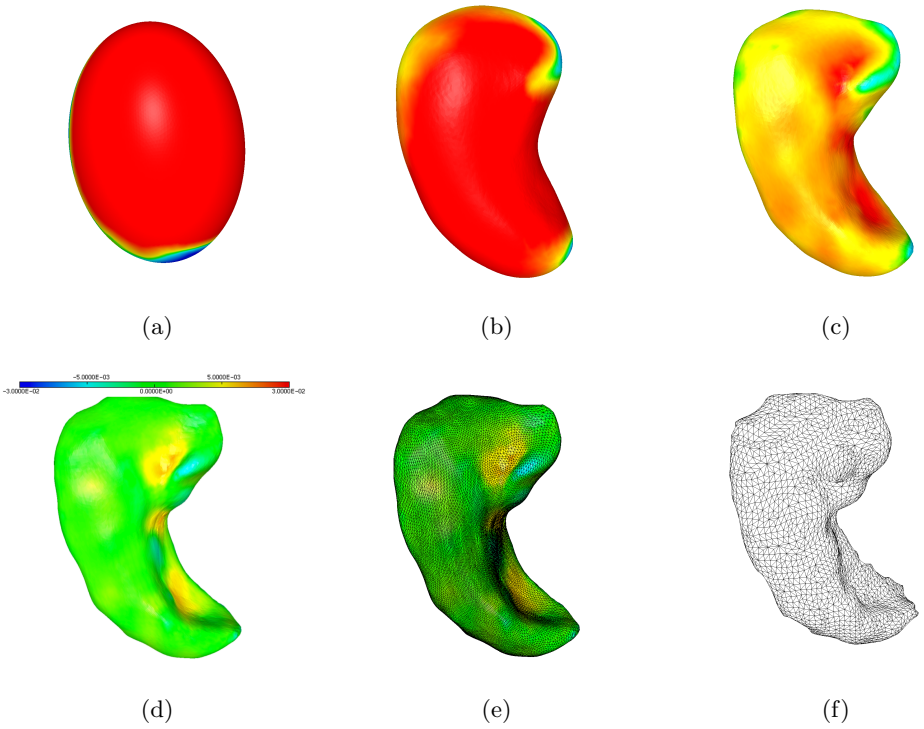


Figure 9: An example in 3D: (a) Template shape Ω_0. (b) Deformed shape Ω_k for $k = 50$. (c) Deformed shape Ω_k for $k = 200$. (d)-(e) Deformed shape Ω_k for $k = 1\,700$. (f) Target shape.

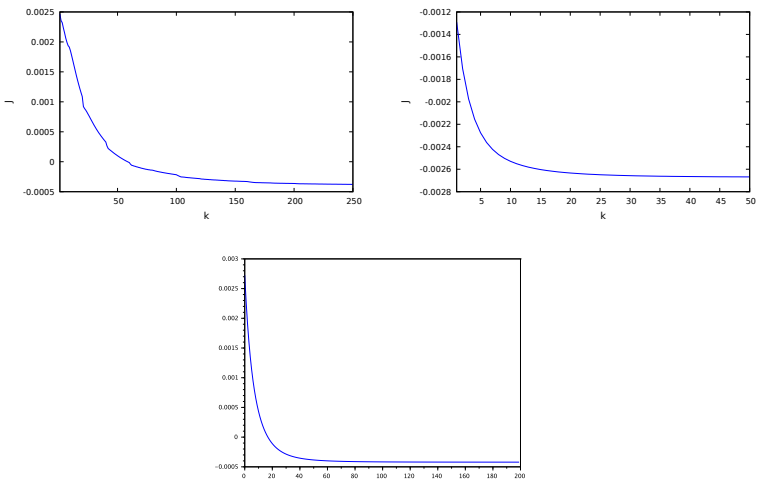


Figure 10: Convergence history for the $3d$ test cases: objective functional $J(\Omega_k)$ versus number of iterations k. Top left: example of Figure 9. Top right: example of Figure 8. Bottom: example of Figure 7.

Using again Theorem 1.2.1, the shape derivative of $J(g)$ reads:

$$\forall \widehat{g} \in \mathcal{G}_{ad},\ \ J'(g)(\hat{g}) = \int_{\partial(I+\chi u_{\Omega,g})(\Omega)} d_{\Omega_T} u_{\Omega,\hat{g}} \cdot n \, ds.$$

Note that the above expression involves the evaluation of the solution $u_{\Omega,\hat{g}}$ with respect to the surface load $\hat{g}$. A more practical expression of the derivative is achieved by introducing an opportune adjoint state. In the following Theorem we make use of the tools presented in Section 1.2.2 concerning the derivation of state-constrained functionals.

Theorem 2.5.1. *The shape functional $J(g)$ is Fréchet-differentiable at $g = 0$, and its derivative reads:*

$$\forall \widehat{g} \in \mathcal{G}_{ad},\ \ J'(g)(\widehat{g}) = -\int_{\partial\Omega} \widehat{g} \cdot p_{\Omega,g} \, ds,$$

where $p_{\Omega,g} \in H_0^1(\Omega)^d$ is an adjoint state, solution to the following variational problem:

$$\forall v \in H_0^1(\Omega)^d,\ \ \int_\Omega \sigma(u_\Omega) : e(v) \, dx = -\int_\Gamma d_{\Omega_T} v \cdot (\mathrm{com}(I + \nabla(\chi u_{\Omega,g}))n) \ ds. \quad (2.5.3)$$

Proof. We rely on the method of Céa (Section 1.2.2) to calculate the derivative. This calculation is merely formal insofar as it assumes that the function $g \mapsto u_{\Omega,g}$ is differentiable. Let us introduce the Lagrangian functional $\mathcal{L} : \mathcal{G}_{ad} \times (H_0^1(\mathbb{R}^d)^d \cap \mathcal{D}(\mathbb{R}^d)^d)^2 \mapsto \mathbb{R}$ defined as:

$$\mathcal{L}(g, v, q) = \int_{(I+\chi v)(\Omega)} d_{\Omega_T} \, dx + \int_\Omega \sigma(u_\Omega) : e(v) \, dx - \int_{\partial\Omega} g \cdot q \, ds.$$

By definition of $u_{\Omega,g}$, one has, for any test function $q \in H_0^1(\mathbb{R}^d)^d \cap \mathcal{D}(\mathbb{R}^d)^d$:

$$\mathcal{L}(g, u_{\Omega,g}, q) = J(g). \quad (2.5.4)$$

For a given load case $g \in \mathcal{G}_{ad}$, we search for the stationary points (u, p) of $\mathcal{L}(g, \cdot, \cdot)$.

- Equating the partial derivative $\frac{\partial \mathcal{L}}{\partial p}(g, u, p)(q)$ to 0 yields:

  $$\forall q \in H^1_{\Gamma_D}(\mathbb{R}^d)^d \cap \mathcal{D}(\mathbb{R}^d)^d,\ \ \int_\Omega \sigma(u_\Omega) : e(q) \, dx - \int_{\partial\Omega} g \cdot q \, ds = 0.$$

  By density of $\mathcal{D}(\mathbb{R}^d)^d$ in $H_0^1(\mathbb{R}^d)^d$, this equality actually holds for $q \in H_0^1(\mathbb{R}^d)^d$, and we readily identify u as $u_{\Omega,g}$, the solution of (1.5.2) with $f = 0$.

- The partial derivative $\frac{\partial \mathcal{L}}{\partial u}(g,u,p)$ reads, for any $v \in H_0^1(\mathbb{R}^d)^d \cap \mathcal{D}(\mathbb{R}^d)^d$:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial u}(g,u,p)(v) &= \int_{\partial(I+\chi u)(\Omega)} d_{\Omega_T}\, v \cdot n_{\Omega_{\chi u_{\Omega,g}}}\, ds + \int_\Omega \sigma(\widehat{v}) : e(p)\, dx \\ &= \int_{\partial\Omega} d_{\Omega_T}\, v \cdot \left(|\mathrm{com}(I+\nabla(\chi u_{\Omega,g}))n|\, n_{\Omega_{\chi u_{\Omega,g}}} \right) ds + \int_\Omega \sigma(v) : e(p)\, dx.\end{aligned}$$

  In the last row, com(A) stands for the matrix of cofactors of A. Now, since $u = u_{\Omega,g}$ is smooth enough, equating this last expression to 0 yields that $p = p_{\Omega,g}$, the unique solution to the adjoint problem (2.5.3).

Eventually, differentiating with respect to g in (2.5.4) yields:

$$\forall \widehat{g} \in \mathcal{G}_{ad},\ \ J'(g)(\widehat{g}) = \frac{\partial \mathcal{L}}{\partial g}(g, u_{\Omega,g}, p)(\widehat{g}) + \frac{\partial \mathcal{L}}{\partial u}(g, u_{\Omega,g}, p)\left(\frac{\partial u_{\Omega,g}}{\partial g}(\widehat{g})\right).$$

Now evaluating at $p = p_{\Omega,g}$, we end up with:

$$\forall \widehat{g} \in \mathcal{G}_{ad},\ \ J'(g)(\widehat{g}) = \frac{\partial \mathcal{L}}{\partial g}(g, u_{\Omega,g}, p_{\Omega,g})(\widehat{g}),$$

which is the desired formula. □

Remark 2.5.2. *In the present setting, the existence result (2.2.1) is no longer guaranteed for any pairs of shapes Ω_0, Ω_T. However, under the additional assumption that Ω_T belongs to the admissible set:*

$$\mathcal{O}_{ad} = \{\Omega_{ad} \subset \mathbb{R}^3 \mid \exists g \in \mathcal{G}_{ad} : u_g(\Omega_0) = \Omega_{ad}\},$$

the optimal load g^* realizes the desired deformation:

$$(I + u_g^*)(\Omega_0) = \Omega_T. \tag{2.5.5}$$

Note that the optimal load g^* and the associated the displacement u_{g^*} realizing equality (2.5.5) are not uniques.

The elastic energy stored during the deformation is written in terms of the optimal displacement field u_g^*:

$$E_{u_{g^*}}(\Omega_0;\Omega_T) = \frac{1}{2}\int_{\Omega_{u_{g^*}}} |\sigma(u_{g^*})|^2 dx.$$

The above energy quantifies the mechanical cost of the deformation form Ω_0 and Ω_T and can be thought as a similarity measure between target and template shape.

Remark 2.5.3. *From the numerical point of view, the identification of a subset Γ_D at which the shapes are clamped guarantees the well-posedness of the elastic problem.*

This condition could be replaced by adding a 0^{th}-order term. For example, imagine that two subsets Σ_T, Σ_0 have been identified respectively on $\partial\Omega_T, \partial\Omega_0$. Then we can cast the minimization problem over the space:

$$H^1_{\Sigma_0}(\Omega)^d := \{v \in H^1(\Omega)^d, v = u_0 \text{ in } \Sigma_0\},$$

where u_0 is a prescribed mapping between Σ_0 and Σ_T.

2.5.1 Numerical issues

From the numerical point of view, again the template shape Ω_0 is discretized as a simplicial mesh (i.e. a triangulation), and the target shape Ω_T is supplied through its signed distance function. Starting from the template shape Ω_0 we perform a gradient descent algorithm with adaptive step size in order to get a sequence of pairs (g_k, u_{g_k}) of surface loads and corresponding displacements with decreasing values of $J(g_k)$. The algorithm stops when the step size is smaller than a fixed tolerance ε. The general procedure is described by the following steps:

Start from an initial shape Ω_0,
for $k = 0, \cdots$ **convergence**

1. compute the adjoint state p_{Ω_k} of the considered PDE system on Ω_k by FEM discretization;
2. compute the state u_{Ω_k} of the considered PDE system on Ω_k by FEM discretization;
3. compute the shape gradient $J'(\Omega_k)$ and infer a descent direction θ_k for the shape functional;
4. perform a line search for getting the optimal step size τ_k;
5. advect the surface load g_k according to:

$$g_{k+1} = g_k + \tau_k g_k.$$

Note that in this context the computation of a descent direction (which is performed at each iteration) involves the evaluation of both direct and adjoint state, leading to the computation of two Finite Elements solutions. More precisely, the two system share the same mass matrix (since they are evaluated on the same coordinate vertices with unchanged connectivity) but the second hand and the resolution of the linear system have to be updated. This makes the computational cost more involved when comparing with the algorithm described in the previous section.

2.5.2 Numerical examples

First, we propose a $2d$ test case. A template shape is initially deformed onto a target shape under the effect of a prescribed surface load (Figure 11 (a)-(b)). Indeed in this case we are by construction in the hypothesis of Remark (2.5.2). Then we run our shape matching algorithm for recovering the surface load that generated the deformation of the template shape. Results are presented in Figure 11 (c)-(d)). The convergence history is reported in Figure 11-(e). The convergence of the procedure is achieved in $k = 1\,100$ iterations for a tolerance $\varepsilon = 1.e^{-6}$. The discrepancy $\mathcal{D}(\partial\partial\Omega_k, \partial\Omega_T)$ calculated on the boundary $\partial\Omega_k$ of the resulting shape

$$\Omega_k = (I + u_{g_k})\,(\Omega_0),$$

equals $8.3e^{-4}$.

Next, we propose a $3d$ test case. As in the previous example, we deform a computational mesh of the template shape Ω_0 under the effect of a prescribed surface load (Figure 12 (a)-(b)). and we recover its deformation by our algorithm. Results are presented in Figure 12-(c)-(d). The convergence of the procedure is achieved in $k = 1\,800$ iterations for a tolerance $\varepsilon = 1.e^{-6}$. The discrepancy $\mathcal{D}(\partial\Omega_k, \partial\Omega_T)$ calculated on the boundary $\partial\Omega_k$ of the resulting domain equals $7.1e^{-4}$.

Comparing the convergence curves of the present approach with the one of the previous section, we remark that the second method is more difficult to achieve convergence. This is because when looking for a global displacement the minimization process try to rotate the object in all direction, making slower the convergence of the overall algorithm. Moreover, the theoretical restriction given in Remak 2.5.2 is very limiting also from the numerical point of view. We experimented the method in a lot of numerical cases and we always observed a slow or a non convergence. We exploited some strategies for overcoming this issue. However, at the moment of the redaction no one was satisfactory. We believe that the proposed setting will be more amenable with a non linear model accounting for large deformations.

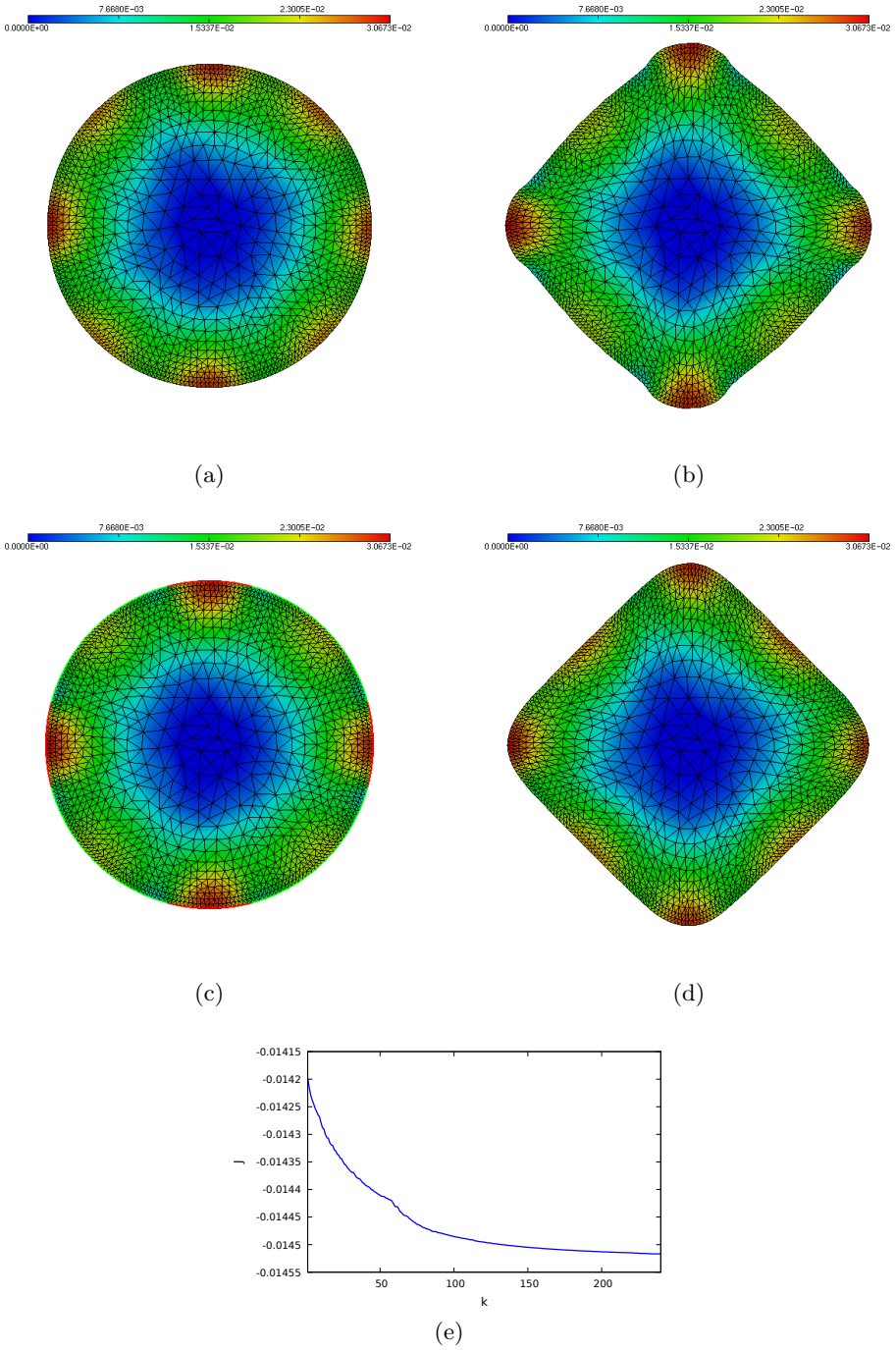


Figure 11: (a) Template shape and color map of the displacement field u_g. (b) Template shape deformed onto the target shape under the displacement u_g. (c)-(d) Recovering of u_g by the algorithm described in Section 2.5.1. (e) Objective functional $J(\Omega_k)$ versus number of iterations k.

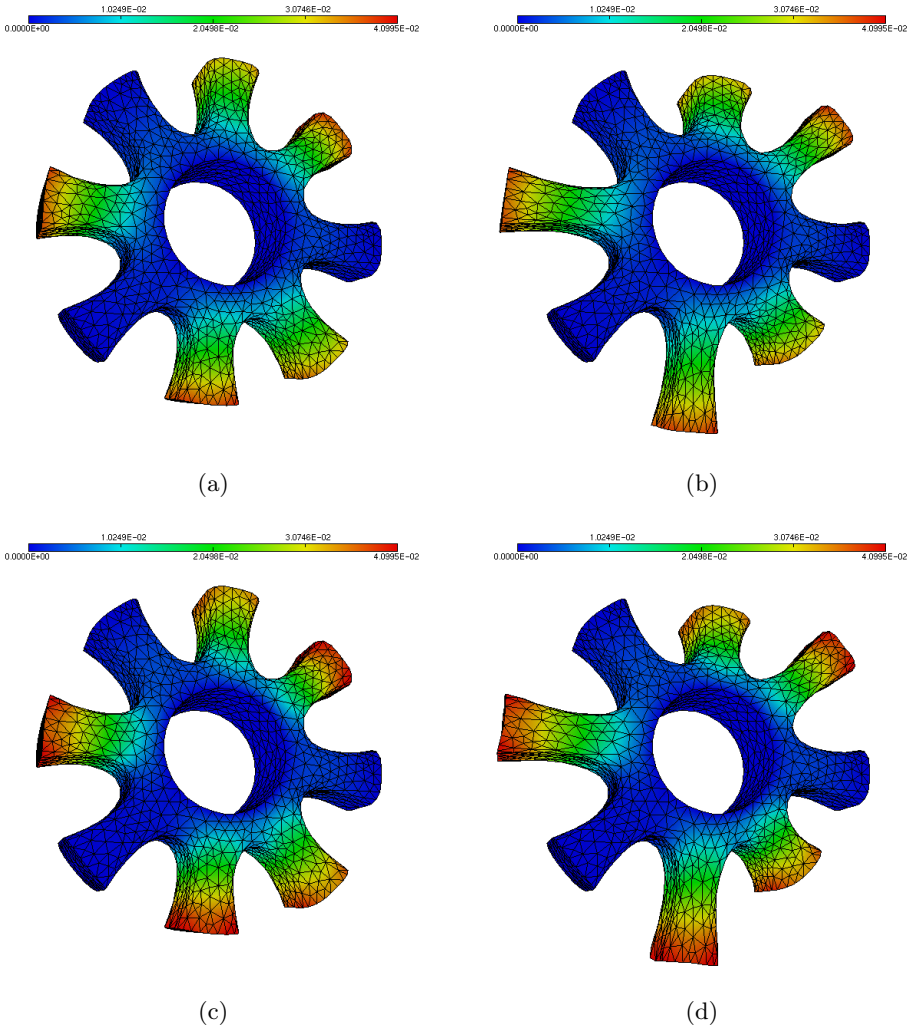


Figure 12: (a) Template shape and color map of the displacement field u_g. (b) Template shape deformed onto the target shape under the displacement u_g. (c)-(d) Recovering of u_g by the algorithm described in Section 2.5.1.

2.6 A boundary functional for shape matching

The purpose of this section is to compare the elastic matching method based on the minimization of the functional featured in (2.2.1) with a more classical approach. Let $\Omega_0, \Omega_T \subset \mathbb{R}^d$ be again respectively 'template' and 'target' shapes. Under the assumption that they share the same topology, we want to map Ω_0 onto Ω_T by minimizing the discrepancy between the boundaries $\partial\Omega_0, \partial\Omega_T$. The dissimilarity between a reference shape Ω and the target shape Ω_T is thus measured by the following

functional $P(\Omega)$ of the domain:

$$P(\Omega) = \int_{\partial\Omega} d^2(x, \partial\Omega_T)(x)\, ds, \tag{2.6.1}$$

where $d(\cdot, \partial\Omega_T)$ denotes the usual Euclidean distance function to $\partial\Omega_T$. This functional is based on a more classical paradigm -aligning two shapes by aligning their boundaries- and is used in a wide variety of situations in the context of non rigid registration. Our purpose is to show that this functional can induce ambiguous situations preventing convergence and preservation of the quality of the mesh used as support of the calculation. By Theorem 1.2.2, the functional (2.6.1) is shape differentiable at any (enough) regular shape Ω, and the associated shape derivative reads:

$$\forall \theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d),\ \ P'(\Omega)(\theta) = \int_{\partial\Omega} \big(\frac{\partial d^2}{\partial n} + H d^2\big)\theta \cdot n\, ds,$$

where $H =$ div n is the mean curvature of $\partial\Omega$. In the same fashion of Section 2.3 we can define a formal procedure for the minimization of a functional inferring a descent direction for P. The 'natural' vector field inferred from the expression of the shape derivative is:

$$\theta = \left(\frac{\partial d^2}{\partial n} + H d^2\right) n. \tag{2.6.2}$$

Note that in the present context the vector field θ has not to be extended to the whole domain; the vector (2.6.2) can be used as a guide for displacing the points on the boundary $\partial\Omega$. However, for the sake of comparison, we place ourselves in the same theoretical and numerical setting of Section 2.4 and use elastic equations for parametrizing the descent direction.

2.6.1 Extension-regularization of the descent direction

Following the construction described in Section 2.4, which remains unchanged in the present setting, we can obtain a descent direction for $P(\Omega_k)$ as the unique solution u_Ω belonging to $H^1_\omega(\Omega)^d := \{v \in H^1(\Omega)^d, v = 0 \text{ in } \omega\}$ of the linearized elasticity system:

$$\begin{aligned}\forall v \in H^1_\omega(\Omega)^d,\ \ \int_\Omega \sigma(u_\Omega) : e(v)\, dx &= -P'(\Omega)(v) \\ &= -\int_{\partial\Omega} \big(\frac{\partial d^2}{\partial n} + H d^2\big)\theta \cdot n\, ds.\end{aligned}$$

Starting from Ω_0, we produce a sequence of shapes $(\Omega_k)_{k=0,\dots,}$ which boundaries $\partial\Omega_k$ get closer and closer to $\partial\Omega_T$.

2.6.2 Numerical issues

As far as the numerical setting is concerned, the template shape Ω_0 is discretized as a simplicial mesh (filled with tetrahedra), and the target shape Ω_T is supplied through a contour mesh of its boundary (triangles in $3d$, edges in $2d$).

We refer to Section 2.4.1 for the general steps of the procedure, which remains unchanged. The major difference relies on the calculation of the descent direction, which in the present context demands the following steps:

At iteration k:

1. compute the discrete mean curvature H at each vertex x_i of the advecting boundary $\partial\Omega_k$.
2. compute the normal derivative of the distance function $d = d(\cdot, \partial\Omega_T)$ at each vertex x_i of the advecting boundary $\partial\Omega_k$ by the formula:
$$\frac{\partial d^2}{\partial n} = \nabla(d^2) \cdot n_i = 2(x_i - p_{\Omega_T}(x_i)) \cdot n_i,$$
where $p_{\Omega_T}(x_i)$ stands for the projection of x_i on $\partial\Omega_T$ and n_i stands for the normal at x_i;
3. compute the state u_{Ω_k} of the considered PDE system on Ω_k by FEM discretization.

The numerical implementation requires at each iteration k the computation of the discrete mean curvature of the moving boundary $\partial\Omega_k$. The computation of differential geometry properties of a discrete surface is a active research field and a huge literxwature has been devoted to this topic. We describe below the two methods that we used in our experiments for the computation of the discrete mean curvature, respectively in $2d$ and $3d$ space dimension.

2.6.3 Computation of the discrete curvature

For the $2d$ case we follow the approach described in [34]. Let $\mathcal{T}$ the boundary of a $2d$ mesh and let $\{x_i\}_{i=0,\dots,n}$ the list of points lying on $\mathcal{T}$. Let us suppose that the edges of the boundary has been sorted in such a way that for each vertex x_i the vertex x_{i-1} (resp. x_{i+1}) denotes the vertex placed before (resp. after) x_i, according to a counterclockwise orientation of $\mathcal{T}$. In such a situation the normal vector at point x_i is estimated as the orthogonal of the (normalized) vector $\overrightarrow{x_i x_{i-1}}$. Then the discrete radius of curvature r_i at vertex x_i is given by the formula:

$$\frac{1}{\kappa_i} = r_i = \frac{1}{4}\left(\frac{\overrightarrow{x_i x_{i-1}} \cdot \overrightarrow{x_i x_{i-1}}}{-n(x_i) \cdot \overrightarrow{x_i x_{i-1}}} + \frac{\overrightarrow{x_i x_{i+1}} \cdot \overrightarrow{x_i x_{i+1}}}{-n(x_i) \cdot \overrightarrow{x_i x_{i+1}}}\right).$$

The discrete curvature κ_i at x_i is taken as the inverse of the radius of curvature when the latter is not vanishing, and is set to zero otherwise.

Let us now discuss the $3d$ case. Most of classical definitions from differential geometry of smooth curves and surfaces need to be extended to the case of discrete surfaces. Practically one has to compute approximations of these differential properties (e.g. normal vector, mean curvature) directly from the discrete data. A general idea (see [77]) is to evaluate a discrete differential quantity at a vertex x of the mesh as a spatial average over a neighborhood $N(x)$ of x. The neighborhood of a vertex x of a surface mesh can be defined as:

- the 1-ring of a vertex, i.e. the collection of all the triangles sharing the vertex;
- the barycenter cells, i.e. the closed polygon obtained by connecting the barycenters of all the triangles belonging to the 1-ring of a vertex;
- the Voronoi cells, i.e. the closed polygon obtained by connecting the circumcenters of all the triangles belonging to the 1-ring of a vertex.

The mean curvature H of a surface at a point p can be equivalently defined by the following integral (see [74]):

$$H = \frac{1}{\pi} \int_0^{\pi} \kappa_n(\theta)\, d\theta, \tag{2.6.3}$$

where $\kappa_n(\theta)$ is the normal curvature at p along a direction making an angle θ with a fixed direction. Let $\Delta_{\mathcal{S}}$ be the Laplace Beltrami operator for the surface $\mathcal{S}$. A discrete approximation K_H of $\Delta_{\mathcal{S}}$ is classically computed via the cotangent formula, firstly introduced in [88]. When the operator is applied on coordinates functions, this formula reads:

$$K_H(x_i) = \frac{1}{2\mathcal{A}_i} \sum_{j \in N(i)} (\cot \alpha_{ij} + \cot \beta_{ij})(x_i - x_j), \tag{2.6.4}$$

where the sum is extended over the 1-neighborhood $N(i)$ of the vertex x_i and α_{ij}, β_{ij} are the angles opposite to the edge $x_i x_j$ (see Figure 13). From this expression the scalar (unsigned) mean curvature $H(x_i)$ reads:

$$H(x_i) = \frac{1}{2} ||K_H(x_i)||. \tag{2.6.5}$$

The sign of the discrete mean curvature is then chosen according to the orientation of the normal field. The above formula can be thought as a quadrature of the integral (2.6.3). Indeed a straightforward calculation (see [77]) shows:

$$H(x_i) = \sum_{j \in N(i)} w_{ij} h^N_{i,j},$$

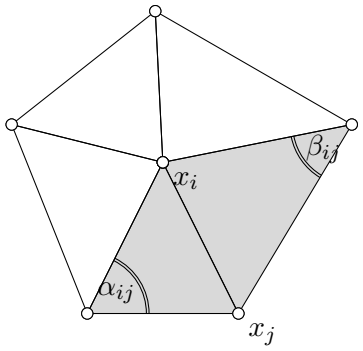


Figure 13: Angles α_{ij} and β_{ij} opposites to the mesh edge $x_i x_j$.

where the weights w_{ij} are given by:

$$w_{ij} = \frac{1}{8\mathcal{A}_i} \left(\cot \alpha_{ij} + \cot \beta_{ij}\right) ||x_i - x_j||^2,$$

and the coefficients $h^N_{i,j}$ are projections of the normal curvature in the direction of $x_i x_j$. Figure 14 shows two examples of computation of the mean curvature distribution. Note that the formula is very sensitive to noise when the mesh under considerations is not regular enough (which is the case for example of models coming from scans segmentation).

2.6.4 Numerical examples

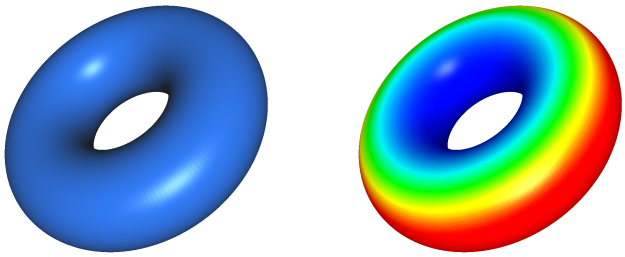
In this section we show three examples of boundary matching.

First, consider the $2d$ test case of Figure 15. The template shape is a small disk located at the interior of a target silhouette. The algorithms ends after $k = 200$, revealing an excellent matching. Note the recovering of the two singularities of the target shape.

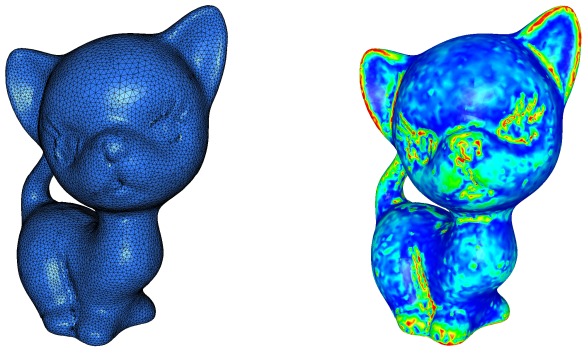
Next, we perform the same test case of Figure 4-(d)-(f). Note that in this context the algorithms fails to convergence, invalidating the mesh after few iterations. The occurrence of singularities is shown in Figure 17.

Also, consider the $3d$ test case of Figure 16, in which a spherical template is mapped onto a cube. The template shape is able to recover the flat regions of the template boundary, but fails in approaching its edges, generating very stretched triangle elements.

Remark 2.6.1. *The occurrence of singularities (as in Figure 17) is due to the non differentiability of the signed distance function (Prop. 1.3.1) when the set of projections of a point is not a singleton. In such case the shape derivative is not well defined since we have as many directions as projection points for advecting the points.*

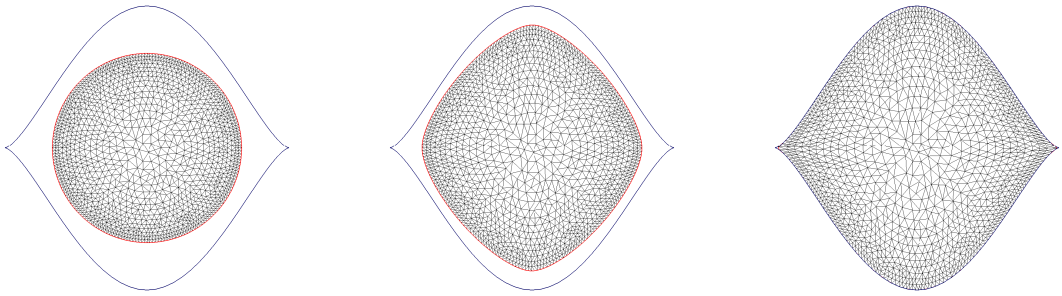


(a)



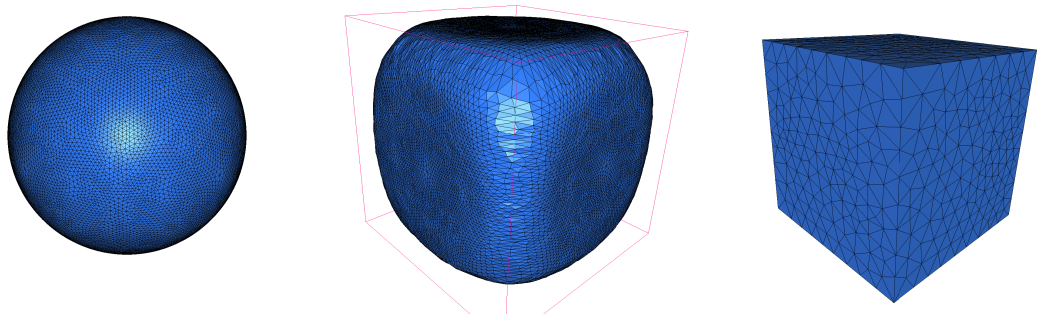
(b)

Figure 14: Mean curvature plot using pseudo colors. (a) Parametric surface test case. (b) Irregular mesh test case.



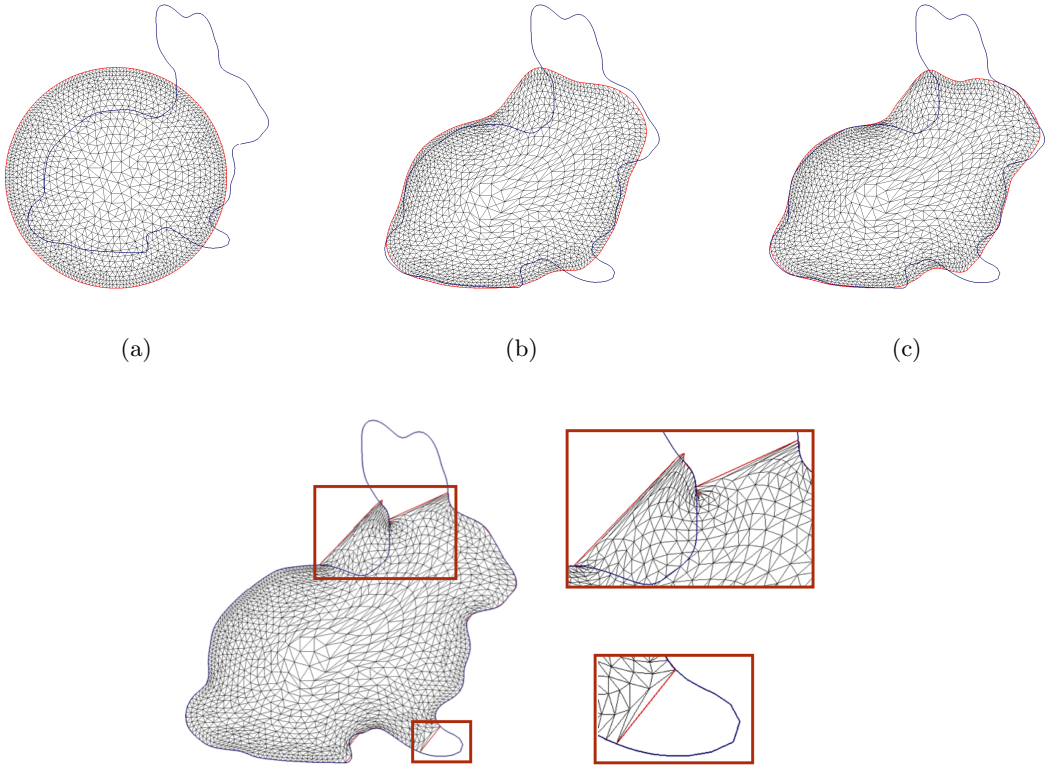
(a) (b) (c)

Figure 15: Computational mesh of the shape Ω_k and boundary of the target shape (blue contour) at iteration (a) $k = 0$; (b) $k = 40$;(c) $k = 200$.



(a) (b) (c)

Figure 16: (a) Computational mesh of the template shape. (b) Deformed template shape showing stretched elements. (c) Target shape.



(a) (b) (c)

(d)

Figure 17: Computational mesh of the shape Ω_k and boundary of the target shape (blue contour) at iteration (a) $k = 0$; (b) $k = 40$;(c) $k = 60$. (d) Deformed shape Ω_{100} generating singularities.

2.7 Conclusive remarks

In this chapter we proposed a novel method for elastic shape matching borrowing techniques from shape optimization. The method has a strong mathematical background, in particular it relies on the existence of a unique solution of a variational problem. The advantages of our method are the simple implementation and the intuitive formulation. The algorithm presented in Section 2.4 has revealed more suitable fro numerical practice with respect to the one featured in Section2.5. The method proposed in Section 2.5 suffers of at least two major limitations. First, for guaranteeing the existence of an optimum, the target shape is imposed to belong to the set of shapes that are recovered by applying surface loads on the template shape. This condition is too restrictive due to the mechanical constraints of linear elasticity. In the case in which the target shape does not belong to the set defined in (2.5.2), the algorithm converges to the best approximation in $\mathcal{O}_{ad}$ of the target shape. In such case the convergence of the algorithm does not guarantees the proximity between the deformed template shape and the target shape. Second, even when the condition on the target shape is satisfied, the numerical implementation shows slow convergence due to the rotational component of the deformation. The method featured in Section 2.4 which accounts for 'small' displacements has revealed to perform quite better in numerical practice -helping the algorithm to escape local minima- and less restrictive in terms of the displacement which can be reached. In particular the method allows to morph a couple of shapes which are not close from one other.

The volumetric functional based on the paradigm 'ask to Ω_0 to occupy all the space of Ω_T' has proven much more efficient with respect the one based on the alignment of boundaries. However, one could think about a combination of the two functionals J and P for dealing with the problem of partial matching. In this context ,let Ω_0, Ω_T be again respectively target and template shapes. Consider a subset $\Sigma_T \subset \partial\Omega_T$, and assume that a corresponding subset $\Sigma_0 \subset \partial\Omega_0$ has been identified on the boundary of Ω_0, which we would like to map to Σ_T. The additional information comes for instance from an automatic mesh segmentation algorithm. Starting from Ω_0, it seems natural to try to minimize the weighted functional:

$$L_\alpha(\Omega) = \int_\Omega d_{\Omega_T}\, dx + \alpha \int_\Sigma d(x, \Sigma_T)\, ds,$$

which impose to the subset Σ_0 to be mapped onto Σ_T.

Chapter 3

Around an existence problem in shape optimization

Contents

3.1 Introduction

This chapter is devoted to the study of an existence problem for a shape functional which arises in the context of structural optimization. At the time of writing we don't have a precise answer to this existence question. We introduce the problem and some theoretical tools to study existence in the context of shape optimization. The proof of a precise statement is still underway.

In Chapter 2 we introduced a shape matching method based on the distance functional:

$$J(\Omega) = \int_{\Omega} d_{\Omega_T}(x)dx,$$

with d_{Ω_T} the signed distance function to a target shape Ω_T. Evaluating the functional J at a certain region of the ambient space gives informations of type 'How much am

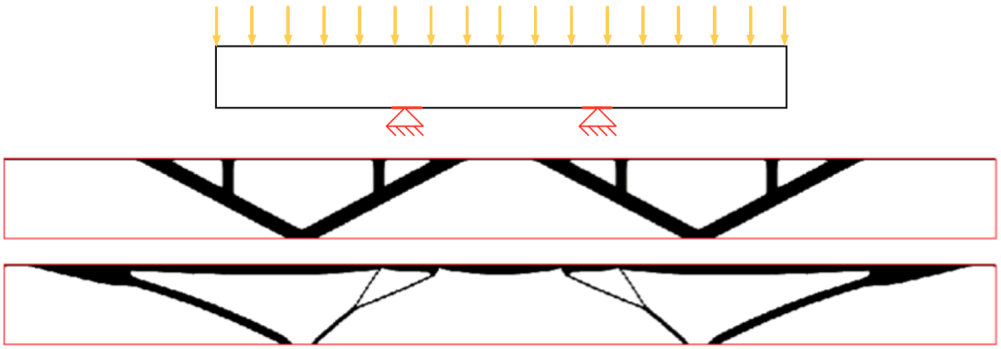


Figure 1: Reprinted from [32]. Top: load case. Optimal shape for problem (3.1.1) for $\gamma = 1$ (middle) and $\gamma = 0.25$ (bottom).

I close to the target shape Ω_T?' or 'Am I at the interior or the exterior of the target domain Ω_T?'. Therefore, the computation of J at a certain part Ω of the ambient space gives informations about the 'closeness to the specific design Ω_T'. Beyond the specific context of shape matching, it seems natural to employ this functional in the more general context of structural optimization and in architectural design oriented applications. Although shape optimization methods are currently ubiquitous in mechanical engineering, the application of their instruments to architectural design is a new and emergency field [32]. The general purpose in architecture oriented shape optimization problems is to incorporate aesthetic criteria -as for example the personal taste of the architect- into a more classical problem which evaluates for instance the mechanical performance of a structure. In [32] several numerical examples of shape optimization problems under geometrical constraints are presented. The functionals under consideration share the very general form $C_m(\Omega) + C_g(\Omega)$, where C_m is the mechanical contribution aiming at guaranteeing the robustness of the structure, and C_g is a geometric constraint, incorporating design aesthetic criteria. In the present context we focus on a specific problem in which the mechanical performance of the shape is evaluated in terms of the 'elastic compliance' and the aesthetic criteria is mathematically formulated in terms of the distance functional J, related to some user-defined specific design Ω_T. In practice, the compliance is evaluated with respect to the solution of a mechanical system. For example, consider the linear elasticity setting introduced in Section 1.5. Consider shapes that are clamped at a part Γ_D of their boundary and subjected to a surface load g on a part $\Gamma_N := \Gamma \setminus \Gamma_D$ of $\Gamma = \partial\Omega$. For a given load case g, consider the solution u_Ω of Problem (1.5.2) posed on Ω. In this situation the compliance $C(\Omega)$ is defined as the work of external loads, namely the quantity:

$$C(\Omega) := \int_\Omega \sigma(u_\Omega) : \varepsilon(u_\Omega)\, dx = \int_{\Gamma_N} g u_\Omega\, ds.$$

Let D be a domain in $\mathbb{R}^2$ and let $\Omega_T \subset D$ be a given target shape. Incorporating the

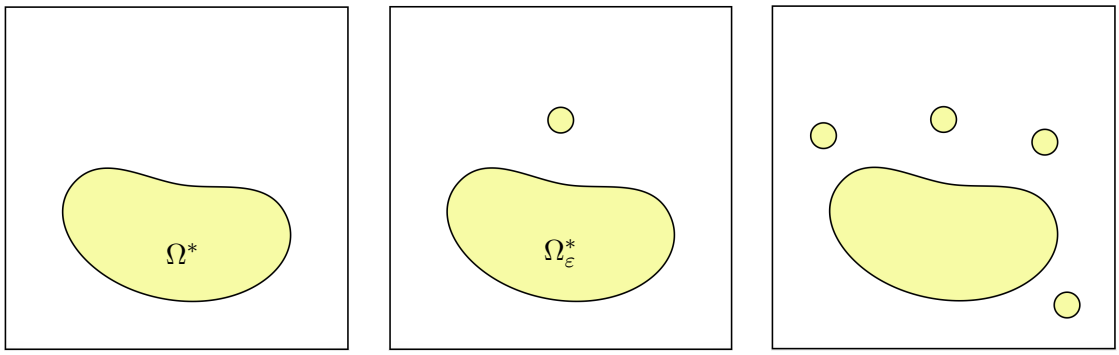


Figure 2: Non existence of an optimal design for Problem (3.2.1).

functional J into the process results in the following minimization problem:

$$\min_{\Omega\in\mathcal{O}_V} S_\gamma(\Omega), \text{ where } S_\gamma(\Omega) = \gamma C(\Omega) + (1-\gamma)J(\Omega), \tag{3.1.1}$$

where $\gamma \in [0,1]$ is a weighting coefficient. The set $\mathcal{O}_V$ of admissible shapes is defined as:

$$\mathcal{O}_V = \{\Omega \subset D, \quad \Omega \text{ open }, \quad |\Omega| = V\}.$$

Figure 1 depicts the numerical computation of an optimal shape for Problem (3.1.1) with ($\gamma = 0.25$) and without ($\gamma = 1$) incorporating the geometric constraint. Note that Problem (3.1.1) admits a unique solution equals to the target shape Ω_T when $\gamma = 0$. When $\gamma = 1$, the reader familiar with theoretical shape optimization problems will recognize a well known example of non existence of optimal shapes: the compliance minimization problem under volume constraint. The non existence of a classical solution (i.e. a black/white design) is due to the occurrence of homogenization phenomena and is a major difficulty in many shape optimization problems. Let us start our presentation with a very simple example of such a phenomenon.

3.2 An introductory example

This examples is due to G. Buttazzo and extracted from [2]. Let $B = B(0,1)$ be the unit ball in $\mathbb{R}^2$. Consider the following least squares minimization problem:

$$\min_{\Omega\in\mathcal{O}} F(\Omega), \text{ where } F(\Omega) := \int_B (u_\Omega - c)^2, \tag{3.2.1}$$

with $\mathcal{O} = \{\Omega \subset B, \Omega \text{ open }\}$ the set of admissible shapes, $c \in \mathbb{R}$ a constant value and the state u_Ω solution of the following Poisson-Dirichlet problem:

$$\begin{cases} -\Delta u_\Omega = 1 & \text{in } \Omega, \\ u_\Omega = 0 & \text{in } B \setminus \Omega. \end{cases} \tag{3.2.2}$$

For instance, one could think about a conductivity problem in which one wants to optimize the location of a mixture of two materials (Ω and $B \setminus \Omega$ stand respectively for the conductive and non conductive material) for matching an ideal constant temperature c. Let us show that Problem (3.2.1) has no solution over $\mathcal{O}$. First, it is easy to see that the whole ball B cannot be a minimizer of F for c small enough. Indeed, consider a ball $B(0,R)$ of ray $0 < R \leq 1$ centered at the origin. It is straightforward to verify that the solution $u_R := u_{B(0,R)}$ of problem (3.2.2) with $\Omega = B(0,R)$ has the following expression:

$$u_R(x) = \begin{cases} \dfrac{R^2 - |x|^2}{4} & \text{if } 0 \leq x \leq R, \\ 0 & \text{if } R < x \leq 1. \end{cases} \tag{3.2.3}$$

Consider the solutions u_1 and u_r, related respectively to the domains B and $B(0,r)$, with $0 < r < 1$. From (3.2.3) we get the inequality:

$$(u_r - c)^2 < (u_1 - c)^2,$$

which holds if c verifies $c < \frac{r^2+1}{8}$. The above inequality leads to:

$$F(B(0,r)) < F(B),$$

which holds for each couple (r,c) with $0 < r < 1$ and c small enough. This shows that B is not optimal. Let us now show that F cannot have a (regular) minimizer $\Omega^* \subsetneq B$. Let us argue by contradiction and suppose that there exists a regular domain $\Omega^* \subsetneq B$ which is a minimizer for F. Since $\Omega^* \subsetneq B$ we can find a point x_0 and a ball centered in x_0 of small ray ε which is strictly contained in the complementary $B \setminus \Omega^*$. Let $B(x_0,\varepsilon)$ such a ball and let $\Omega^*_\varepsilon = \Omega^* \cup B(x_0,\varepsilon)$ (see Figure 2). Since Ω^*_ε is composed of two disjoints components and the solution of (3.2.2) can be explicitly computed for spherical domains, we can easily write the state $u_{\Omega^*_\varepsilon}$ in terms of u_{Ω^*}:

$$u_{\Omega^*_\varepsilon}(x) = \begin{cases} u_{\Omega^*}(x) & \text{if } x \in B \setminus B(x_0,\varepsilon), \\ \dfrac{\varepsilon^2 - |x - x_0|^2}{4} & \text{if } x \in B(x_0,\varepsilon). \end{cases}$$

We can now compare the cost F at Ω^*_ε and Ω^*:

$$\begin{aligned} F(\Omega^*_\varepsilon) &= \int_B \left(u_{\Omega^*_\varepsilon} - c\right)^2 dx \\ &= \int_{\Omega^*_\varepsilon} \left(u_{\Omega^*_\varepsilon} - c\right)^2 dx + \int_{B\setminus\Omega^*_\varepsilon} \left(u_{\Omega^*_\varepsilon} - c\right)^2 dx \\ &= \int_{\Omega^*} (u_{\Omega^*} - c)^2 \, dx + \int_{B(x_0,\varepsilon)} \left(u_{\Omega^*_\varepsilon} - c\right)^2 dx + \int_{B\setminus\Omega^*} (u_{\Omega^*} - c)^2 \, dx - \int_{B(x_0,\varepsilon)} c^2 \, dx \\ &= F(\Omega^*) + \int_{B(x_0,\varepsilon)} \left(\left(\frac{\varepsilon^2 - |x - x_0|^2}{4} - c \right)^2 - c^2 \right) dx. \end{aligned}$$

Since from (3.2.3) we get $0 < u_{\Omega^*_\varepsilon} < c$ on $B(x_0, \varepsilon)$ for ε small enough, we end up with $F(\Omega^*_\varepsilon) < F(\Omega^*)$, proving that Ω^* is not a minimizer for F. Let us observe that we can iterate this procedure by adding more small components to Ω^*, always getting a better value of F (see Figure 2). This behaviour shows that the temperature c is approached by an homogenization process.

3.3 The compliance minimization problem

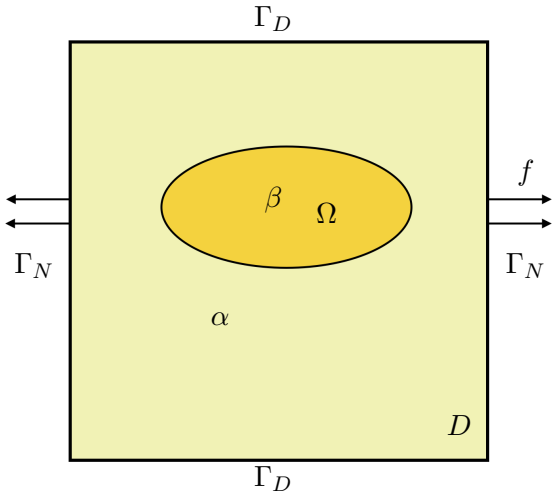


Figure 3: Model problem for Section 3.3.

Before getting things more rigorous, let us now show how an homogenization problem prevents the existence of optimal shapes in the case of the compliance minimization.

Let $D = [0,1]^2$ be a fixed design space and consider the set $\mathcal{O}_V$ of all open measurable sets contained in D of fixed volume V. Consider the space

$$\mathcal{U}_V = \left\{ \chi \in L^\infty(D), \chi(1-\chi) = 0, \int_D \chi(x)dx = V \right\}.$$

of characteristic functions of elements of $\mathcal{O}_V$. We consider the model of an elastic membrane supported on D which is submitted to a constant-valued surface load $f = e_1$ on a part of the boundary $\Gamma_N := \Gamma_0 \cup \Gamma_1$, with $\Gamma_0 = \{0\} \times [0,1]$ and $\Gamma_1 = \{1\} \times [0,1]$. Also, the membrane is clamped at a part $\Gamma_D := \partial D \setminus \Gamma_N$ of its boundary. Let $0 < \alpha < \beta$ be two real numbers. Suppose the elastic membrane to be composed of a mixture of two materials (soft and hard) of elastic coefficients respectively α, β. For all $\Omega \in \mathcal{O}_V$, we define the global coefficient of the membrane as:

$$a_\chi = \alpha\chi_{D\setminus\Omega} + \beta\chi,$$

where χ is the characteristic function of Ω. The membrane is governed by the fol-

lowing elasticity system:

$$\begin{cases} -\mathrm{div}(a_\chi \nabla u_\chi) = 0, & \text{in } D, \\ a_\chi \nabla u_\chi \cdot n = f, & \text{on } \Gamma_N, \\ u_\chi = 0, & \text{on } \Gamma_D. \end{cases} \tag{3.3.1}$$

For each $\Omega \subset D$ Problem (3.3.1) admits a unique solution $u_\chi \in H^1_{\Gamma_D}(D)$ which is the minimizer of the following energy functional:

$$E(u_\chi) = \frac{1}{2}\int_D a_\chi |\nabla u_\chi|^2 dx - \int_{\Gamma_N} e_1 u_\chi ds.$$

We introduce the elastic compliance

$$C(\chi) = -2E(u_\chi) = \int_{\Gamma_N} f u_\chi ds = \int_D a_\chi |\nabla u_\chi|^2 dx, \tag{3.3.2}$$

and we are interested in the constrained minimization problem:

$$\inf_{\chi \in \mathcal{U}_V} C(\chi). \tag{3.3.3}$$

The non existence result extracted from [1] is the following.

Theorem 3.3.1. *If $0 < V < 1$ then problem (3.3.3) has no solution over $\mathcal{U}_V$.*

The complete proof of the theorem can be found in [1]. Here we limit ourselves to report some points of the proof from which we can deduce some properties of the compliance functional. For finding a lower bound m for $C(\chi)$, showing therefore that the quantity $\inf_{\chi \in \mathcal{U}_V} C(\chi)$ is finite, one can rely on the following duality lemma.

Lemma 3.3.1. *Let u_χ be the solution of Problem* (3.3.1)*. Then the stress $\sigma_\chi = a_\chi \nabla u_\chi$ is the minimum of the following energy functional:*

$$\sigma_\chi = \arg\min_{\sigma \in \mathcal{A}} F(\sigma), \text{ where } F(\sigma) := \frac{1}{2}\int_\Omega a_\chi^{-1} |\sigma|^2.$$

The space $\mathcal{A}$ is the set of admissible stresses:

$$\mathcal{A} := \left\{ \sigma \in L^2(\Omega)^2 \text{ such that } -div\, \sigma = 0, \sigma \cdot n = e_1 \text{ on } \Gamma_N, \sigma = 0 \text{ on } \Gamma_D \right\}.$$

Using the above lemma Problem (3.3.3) can be rewritten as:

$$\inf_{\chi \in \mathcal{U}_V} \inf_{\sigma \in \mathcal{A}} \int_D a_\chi^{-1} \sigma \cdot \sigma \, dx.$$

Now, one can show that the function $a_\chi^{-1}\sigma \cdot \sigma$ is convex (see [1]). Using this property

we get the following upper bound for the compliance:

$$\int_D a_\Omega^{-1}\sigma\cdot\sigma\,dx \geq m := (\beta V+\alpha(1-V))^{-1}, \qquad \forall\sigma\in\mathcal{A},$$

which does not depend on Ω. Moreover, it can be shown that the infimum value is not attended in $\mathcal{U}_V$, i.e. that:

$$\forall\chi\in\mathcal{U}_V,\ \ m<C(\chi).$$

Actually, one can prove that the above featured quantity $m := (\beta V+\alpha(1-V))^{-1}$ is exactly the infimum of the compliance functional:

$$\inf_{\chi\in\mathcal{U}_V} C(\chi) = m.$$

This goal is accomplished by exhibiting a sequence of characteristic functions $(\chi_n)_{n\in\mathbb{N}}$ of $\mathcal{U}_V$ which minimizes C and converges to a function which does not belong to $\mathcal{U}_V$. In this precise case a minimizing sequence can be constructed explicitly (see Figure 4). Let us consider the set of characteristic functions $(\chi_n)_{n\in\mathbb{N}}$ defined as follows. Consider the following periodic function:

$$\chi(y) = \begin{cases} 0 & 0\leq y<V,\\ 1 & V\leq y<1,\end{cases}$$

and define the sequence as:

$$\chi_n(x,y) = \chi(ny) \qquad \text{for } n\in\mathbb{N}. \tag{3.3.4}$$

It can be shown using homogenization theory (see [1] for more details) that:

$$\lim_{n\to\infty} C(\chi_n(x,y)) = m,$$

and the sequence $(\chi_n)_{n\in\mathbb{N}}$ does not converges to the characteristic function of a domain. Let us explain the above result: in order to decrease the values of C, the minimizing sequence has to create more and more 'holes' (i.e. has to remove the soft material α) aligned with the horizontal boundary load. The optimum is then reached by a composite material, microscopic mixture of the two original materials. See also [60] for a similar example in the case of a mixture of material and void. Note that the above result reposes on the explicit construction of a minimizing sequence of shapes, which appears a bit miraculous, and no general technique exists to construct such a sequence.

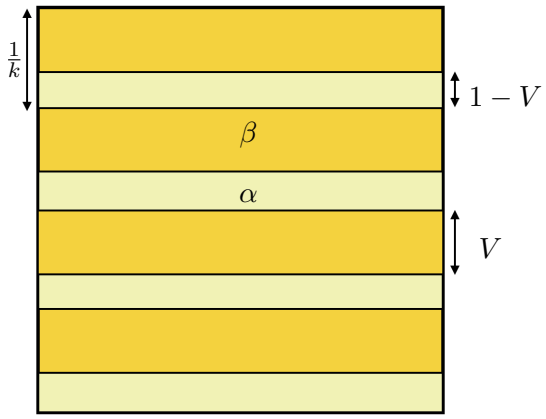


Figure 4: Minimizing sequence for the compliance functional.

3.4 General scheme for getting an existence result

Let us now recall the theoretical requirements for getting an existence result for a shape optimization problem. Let $F(\Omega)$ be a functional of the domain which is minimized over a set $\mathcal{O}_{ad}$ of admissible shapes. Generally speaking, an existence result for a shape minimization problem comes with the verification of the following properties (see [60]):

- the quantity $m := \inf_{\Omega \in \mathcal{O}_{ad}} F(\Omega)$ is finite (boundedness of the functional);
- For each sequence $(\Omega_n)_{n\in\mathbb{N}}$ of shapes in $\mathcal{O}_{ad}$ verfying $\sup_n |F(\Omega_n)| < +\infty$, there exists a shape $\Omega^* \in \mathcal{O}_{ad}$ and a subsequence $(\Omega_{n_k})_{k\in\mathbb{N}}$ which converges (with respect to some topology) to Ω^* (compactness of the functional);
- F is sequentially lower semicontinuous:

$$\text{if } \Omega_n \text{ converges to } \Omega^* \text{ then } J(\Omega^*) \leq m.$$

A crucial and difficult point is the proof of the compactness of the functional. For making this point more rigorous we need to equip the space of admissible shapes with a topology. We will present in the following section two topologies which are usually used in shape optimization context.

3.4.1 Choice of the topology

The choice of the topology on the set of admissible shapes is very important and depends strongly on the nature of the problem at hand. Such a topology should provide both semicontinuity of the objective functional and compactness of the set of admissible shapes. A fruitful approach has been provided in the context of geometric

measure theory by using the strong L^2 topology of the characteristic functions. Let us introduce a weak notion of convergence for characteristic functions over the space $L^\infty(\mathbb{R}^d)$.

Definition 3.4.1. *(Weak-$\star$ convergence) Let $(\Omega_n)_{n\in\mathbb{N}}$ be a sequence of measurable domains with corresponding sequence of characteristic functions $(\chi_{\Omega_n})_{n\in\mathbb{N}}$. The sequence $(\chi_{\Omega_n})_{n\in\mathbb{N}}$ is said to converge to $\chi \in L^\infty(\mathbb{R}^d)$ in the weak-$\star$ topology if:*

$$\forall \psi \in L^1(\mathbb{R}^d),\ \lim_{n\to\infty} \int_{\mathbb{R}^d} \chi_{\Omega_n}\psi = \int_{\mathbb{R}^d} \chi\psi.$$

Definition 3.4.2. *(Convergence in the sense of characteristic functions) Let $(\Omega_n)_{n\in\mathbb{N}}$ be a sequence of measurable sets and let Ω be a measurable set. The sequence $(\Omega_n)_{n\in\mathbb{N}}$ is said to converge to Ω in the sense of characteristic functions if:*

$$\chi_{\Omega_n} \to \chi_\Omega \text{ in } L^p_{loc}(\mathbb{R}^d).$$

Note that this is a topology on a set of equivalence classes: it does not distinguish between shapes that differ only of a set of zero Lebesgue measure. The following result (from [60]) explains the link between weak and strong convergence.

Proposition 3.4.1. *Let $(\Omega_n)_{n\in\mathbb{N}}$ be a sequence of measurable sets of $\mathbb{R}^d$. Then there exists a subsequence $(\Omega_{n_k})_{k\in\mathbb{N}}$ such that $\chi_{\Omega_{n_k}}$ converges weakly-$\star$ in L^∞ to a function χ^* belonging to the space $L^\infty(\mathbb{R}^d, [0,1])$. Moreover, if χ^* is a bang-bang function (i.e. is the characteristic function of a domain) the convergence is strong (i.e. takes place in $L^p(\mathbb{R}^d)$ for some $p \in [1, +\infty]$).*

The problem is that in general the sole weak-$\star$ convergence does not guarantee the limit function to be a characteristic function. The lack of strong convergence is at the basis of homogenization phenomena explored in Sections 3.4 and 3.3. In general, the convergence takes place in the space of density functions:

$$\mathcal{W}_V = \left\{ \chi \in L^\infty(D), 0 \le \chi \le 1, \int_D \chi(x)dx = V \right\}. \tag{3.4.1}$$

See [60] for other topologies used in the context of shape optimization.

The probably most natural way to get extra compactness properties of a set of admissible domains with respect to an objective functional is to restrain the set of admissible domains by adding geometrical, topological or regularity constraints. Such a choice depends strongly on the nature of the functional at hand. In order to get an intuition about the origin of the non existence phenomenon it is useful to analyze the minimizing sequences of the problem at hand. For example, the minimizing sequence given in Section 3.3 fulfills the following properties:

- the shapes do not enjoy Lipschitz regularity of the boundary;

- the perimeter of the shapes is not uniformly bounded;
- the number of their connected components is unbounded.

The first idea coming in mind is to restraint the space in order to prevent shapes to show one of the above behaviors. See Appendix A for a survey of some classical existence results obtained by following this idea.

3.5 Presentation of our problem

We now come to the setting described in Section 3.3. Our aim is to incorporate into the compliance functional the distance functional defined in Chapter 2. We consider the following minimization problem:

$$\min_{\chi\in\mathcal{U}_V} S_\gamma(\chi), \text{where } = S_\gamma(\chi) := \gamma C(\chi) + (1-\gamma)J(\chi), \tag{3.5.1}$$

where $C(\chi)$ and $\mathcal{U}_V$ are the compliance and admissible space defined in Section 3.3, $\gamma \in [0,1]$ is a weighting parameter and $J(\chi)$ is the functional defined in Chapter 2:

$$J(\chi) = \int_D \chi d_{\Omega_T} dx,$$

where Ω_T is a shape in $\mathcal{O}_V$. Note that we can always consider the case of Ω_T of measure V without loss of generality. Indeed, suppose that the measure of Ω_T does not equal V. Then we can define a constant $M \in \mathbb{R}$ and a set

$$\Omega_T^V = \{x \in D, d_{\Omega_T}(x) \leq M\},$$

such that $|\Omega_T^V| = V$. In this case, we can modify the functional S_γ by observing:

$$\arg\min_{\chi\in\mathcal{U}_V} S_\gamma(\chi) = \arg\min_{\chi\in\mathcal{U}_V} \left(\gamma C(\chi) + (1-\gamma)\tilde{J}(\chi)\right),$$

where

$$\begin{aligned}\tilde{J}(\chi) = J(\chi) - MV = &\int_D \chi d_{\Omega_T} dx - M\int_D \chi dx \\ = &\int_D \chi(d_{\Omega_T} - M)dx = \int_D \chi d_{\Omega_T^V} dx.\end{aligned}$$

3.5.1 Boundedness of the functional

The boundedness of the functional is easy to show. For the geometric constraint we have:

$$J(\chi_{\Omega_T}) \leq J(\chi) \leq J(\chi_{D\setminus\Omega_T}), \text{ for each } \chi \in \mathcal{U}_V.$$

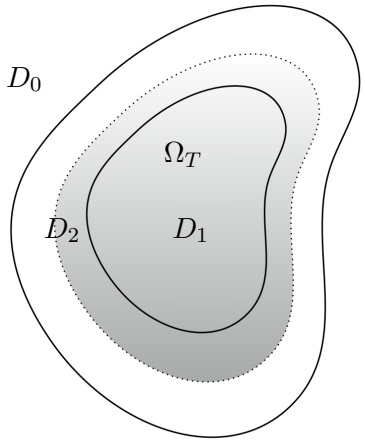


Figure 5: Sets D_0, D_1, D_2 featured in (3.5.2) and target shape Ω_T (grey color).

Moreover, from Section 3.3 we have the following upper bound for the compliance:

$$C(\chi) = \int_D a_\chi^{-1}\sigma \cdot \sigma \, dx \geq (\beta V + \alpha(1-V))^{-1}.$$

Consequently we get the following lower bound m_S for S_γ:

$$m_S := \gamma(\beta V + \alpha(1-V))^{-1} + (1-\gamma)J(\chi_{\Omega_T}).$$

3.5.2 Optimality conditions in $\mathcal{W}_V$

Introduce a sequence $(\chi_n)_{n\in\mathbb{N}}$ of characteristic functions of $\mathcal{U}_V$ which minimizes S_γ. Consider the weak-$\star$ limit χ^* of $(\chi_n)_{n\in\mathbb{N}}$ in $L^\infty(\mathbb{R}^N)$. Following the proof of an existence result in [71], we will now infer optimality conditions for S_γ over the space $\mathcal{W}_V$ of density functions introduced in (3.4.1). Our aim would be to prove that for some value of $\gamma \in (0,1)$ the limit function χ^* is the characteristic function of an optimal domain $\chi^* = \chi_{\Omega^*}$. This property would allow to conclude that χ^* is also the minimizer of S_γ over the space $\mathcal{U}_V$. Let us define the following three sets:

$$D_0 = \{x \in D, \chi^*(x) = 0\}, \quad D_1 = \{x \in D, \chi^*(x) = 1\}, \quad D_2 = \{x \in D, 0 < \chi^*(x) < 1\}. \tag{3.5.2}$$

Note that if $|D_2| = 0$, then χ^* is the characteristic function of a domain and Problem (3.5.1) admits a classical solution; conversely when $|D_2| > 0$ the optimum of (3.5.1) is a density. Let us now characterize the space of admissible variations in the space $\mathcal{W}_V$. Imposing that $\chi^* + th$ belongs to $\mathcal{W}_V$ leads to the following conditions on h:

$$h(x) \geq 0 \text{ on } D_0, \quad h(x) \leq 0 \text{ on } D_1 \quad \text{and} \quad \int_D h(x)dx = 0. \tag{3.5.3}$$

Let us define the space of admissible variations $\mathcal{H}_{ad}$ as the space of all functions $h : D \mapsto D$ verifying the properties featured in (3.5.3).

Theorem 3.5.1. *The functional S_γ admits the following Gâteau derivative (which we call S'_γ with a little abuse of notations):*

$$\forall h \in \mathcal{H}_{ad},\ \ S'_\gamma(\chi^*)(h) = -\gamma \int_D h(x)(\beta-\alpha)|\nabla u_{\chi^*}(x)|^2\,dx + (1-\gamma)\int_D h(x) d_{\Omega_T}(x)\,dx. \tag{3.5.4}$$

Proof. Consider a real parameter $t \in \mathbb{R}^+$ and a function $h(x) \in \mathcal{H}_{ad}$. For the functional J we have:

$$\lim_{t\to 0} \frac{J(\chi^*+th)-J(\chi^*)}{t} = \lim_{t\to 0} \frac{\int_D (\chi^*(x)+th(x)-\chi^*(x)) d_{\Omega_T}(x)\,dx}{t} = \int_D h(x) d_{\Omega_T}(x)\,dx.$$

For the compliance term we have:

$$\lim_{t\to 0} \frac{C(\chi^*+th)-C(\chi^*)}{t} = \lim_{t\to 0} \frac{\int_{\Gamma_N} (u_{\chi^*+th}-u_{\chi^*})\, f\, ds}{t} = \int_{\Gamma_N} f\delta u\, ds,$$

where we used the following asymptotic expansion in the neighborhood of $t=0$:

$$u_{\chi^*+th} = u_{\chi^*} + t\delta u + o(t^2). \tag{3.5.5}$$

The contribution δu in (3.5.5) is defined as the solution of the following linearized problem:

$$\begin{cases} -\mathrm{div}(a_{\chi^*}\nabla\delta u) = \mathrm{div}(h(\beta-\alpha)\nabla u_{\chi^*}), & \text{in } D, \\ a_{\chi^*}\nabla\delta u\cdot n = -h(\beta-\alpha)\nabla u_{\chi^*}\cdot n, & \text{on } \Gamma_N, \\ \delta u = 0, & \text{sur } \Gamma_D. \end{cases} \tag{3.5.6}$$

We introduce an adjoint state p and we integrate by parts (3.5.5) to obtain:

$$\begin{aligned} -\int_D \mathrm{div}(a_{\chi^*}\nabla\delta u)p\,dx &= \int_D \mathrm{div}(h(\beta-\alpha)\nabla u_{\chi^*})p\,dx \\ \int_D a_{\chi^*}\nabla\delta u\cdot\nabla p\,dx - \int_{\Gamma_N} a_{\chi^*}\nabla\delta u\cdot np\,ds &= -\int_D h(\beta-\alpha)\nabla u_{\chi^*}\cdot\nabla p\,dx + \\ &\quad + \int_{\Gamma_N} h(\beta-\alpha)\nabla u_{\chi^*}\cdot np\,ds \\ -\int_D \mathrm{div}(a_{\chi^*}\nabla p)\delta u\,dx + \int_{\Gamma_N} a_{\chi^*}\nabla p\cdot n\delta u\,ds &= -\int_D h(\beta-\alpha)\nabla u_{\chi^*}\cdot\nabla p\,dx. \end{aligned}$$

Therefore, by choosing $p_{\chi^*} = -u_{\chi^*}$, solution of (3.3.1) for a load case equals to $-f$, we end up with:

$$-\int_{\Gamma_N} f\delta u\,ds = \int_D h(\beta-\alpha)|\nabla u_{\chi^*}|^2\,dx.$$

Eventually we get:

$$S'_\gamma(\chi^*)(h) = -\gamma \int_D h(x)(\beta-\alpha)|\nabla u_{\chi^*}(x)|^2\,dx + (1-\gamma)\int_D h(x) d_{\Omega_T}(x)\,dx.$$

□

Since χ^* is a minimizer for S_γ we get the following inequality:

$$S'_\gamma(\chi^*)(h) \geq 0.$$

Using Theorem 3.5.1 we get:

$$S'_\gamma(\chi^*)(h) = -\gamma \int_D h(\beta-\alpha)|\nabla u_{\chi^*}|^2\,dx + (1-\gamma)\int_D h(x) d_{\Omega_T}(x)\,dx \geq 0. \tag{3.5.7}$$

Let us consider two points $x_0, x_1 \in D_2$ and two sequences of open sets $(G_k)_{k\in\mathbb{N}}$ and $(H_k)_{k\in\mathbb{N}} \subset D_2$ converging respectively to x_0 and x_1 and such that $|G_k| = |H_k|$ for each $k \in \mathbb{N}$. Then we choose in (3.5.7) the following sequence $(h_k)_{k\in\mathbb{N}}$:

$$\forall k \in \mathbb{N},\ \ h_k = \chi_{G_k} - \chi_{H_k} \in \mathcal{H}_{ad}.$$

When k goes to $+\infty$ we obtain the following inequality:

$$-\gamma(\beta-\alpha)|\nabla u_{\chi^*}(x_0)|^2 + (1-\gamma)d_{\Omega_T}(x_0) + \gamma(\beta-\alpha)|\nabla u_{\chi^*}(x_1)|^2 - (1-\gamma)d_{\Omega_T}(x_1) \geq 0. \tag{3.5.8}$$

Choosing instead $h_k = -\chi_{G_k} + \chi_{H_k} \in \mathcal{H}_{ad}$ leads to:

$$\gamma(\beta-\alpha)|\nabla u_{\chi^*}(x_0)|^2 - (1-\gamma)d_{\Omega_T}(x_0) - \gamma(\beta-\alpha)|\nabla u_{\chi^*}(x_1)|^2 + (1-\gamma)d_{\Omega_T}(x_1) \geq 0. \tag{3.5.9}$$

From (3.5.8) and (3.5.9) we get an equality for all x_0, x_1 in D_2. We end up with:

$$-\gamma(\beta-\alpha)|\nabla u_{\chi^*}(x)|^2 + (1-\gamma)d_{\Omega_T}(x) = K, \qquad \forall x \in D_2. \tag{3.5.10}$$

where K is a constant value. Note that in this context we cannot conclude $K = 0$ because of the volume constraint. Next, let us choose a point x_0 in D_2 and a point x_1 in D_1. By using the same strategy than above we find the inequality:

$$-\gamma(\beta-\alpha)|\nabla u_{\chi^*}(x)|^2 + (1-\gamma)d_{\Omega_T}(x) \leq K, \qquad \forall x \in D_1. \tag{3.5.11}$$

Note that the constant K depends on the parameter γ, the design space D, the target shape Ω_T and the load case f and it is hard to characterize. Let us distinguish two cases. First, let us suppose $K \geq 0$. In this case, from (3.5.11) we have:

$$d_{\Omega_T}(x) \leq \frac{K + \gamma(\beta-\alpha)|\nabla u_{\chi^*}(x)|^2}{1-\gamma}, \qquad \forall x \in D_1.$$

Therefore inequality (3.5.11) is verified for all $x \in \Omega_T$, so as to get:

$$\Omega_T \subset D_1.$$

Let us argue by contradiction and suppose that $|D_2| > 0$. Then:

$$\int_{D_1 \cup D_2} \chi^* dx > \int_{D_1} \chi^* dx \geq \int_{\Omega_T} dx = V.$$

By construction $\chi^* \in \mathcal{W}_V$, then the volume constraint implies:

$$\int_{D_1 \cup D_2} \chi^* dx = \int_D \chi^* dx = V,$$

which is a contradiction. Note that in such a case we would have $\chi^* = \chi_{\Omega_T}$ for each $\gamma \in (0,1)$, which looks a bit unrealistic. For the case $K = -L$ with $L > 0$, we did not find a contradiction with the assumption $|D_2| > 0$. Then, it is possible in this case the existence of an 'homogenization' zone of non zero measure. Then, it is possible in this case the existence of an 'homogeneization' of non zero measure.

3.5.3 Optimality conditions in $\mathcal{O}_V$

The aim of this section is to infer optimality conditions for S_γ in the space $\mathcal{O}_V$ of open subsets of D of measure V. By a slight change of notations, for each $\Omega \in \mathcal{O}_V$ of characteristic function χ_Ω, we pose $a_\Omega = \alpha(1 - \chi_\Omega) + \beta\chi_\Omega$ and for $f \in H^{-1}(\Gamma_N)$, we consider the following problem:

$$\text{Find } u_\Omega \in H^1_{\Gamma_D}(D) \text{ such that } \quad \begin{cases} -\text{div}(a_\Omega \nabla u_\Omega) = 0, & \text{in } D, \\ a_\Omega \nabla u_\Omega \cdot n = f, & \text{on } \Gamma_N, \\ u_\Omega = 0, & \text{on } \Gamma_D. \end{cases} \tag{3.5.12}$$

Moreover, let us pose:

$$C(\Omega) = \int_{\Gamma_N} f u_\Omega \, ds \text{ and } J(\Omega) = \int_\Omega d_{\Omega_T} \, dx,$$

and consider the minimization problem:

$$\min_{\Omega \in \mathcal{O}_V} S_\gamma(\Omega), \text{ where } S_\gamma(\Omega) = \gamma C(\Omega) + (1-\gamma) J(\Omega).$$

Again, consider a minimizing sequence $(\Omega_n)_{n\in\mathbb{N}}$ for S in $\mathcal{O}_V$. Let χ^* be the limit of χ_{Ω_n} in the weak-$\star$ topology and suppose now that χ^* is the characteristic function of a domain Ω^*. Next theorem gives an expression for the shape derivative of $S_\gamma(\Omega)$. Let us define the space of admissible variations:

$$\Theta_{ad} = \left\{ \theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d) \text{ such that } \quad \text{div}\theta = 0 \right\}.$$

Theorem 3.5.2. *The functional $S_\gamma(\Omega)$ admits a shape derivative at $\theta = 0$ which reads:*

$$\forall \theta \in \Theta_{ad},\ \ S'_\gamma(\Omega^*)(\theta) = \int_{\partial\Omega^*} (-a_{\Omega^*}|\nabla u_{\Omega^*}|^2 + d_{\Omega_T})\theta \cdot n \, ds.$$

Proof. The proof follows from Theorem 1.2.1 in the case with volume constraint. Let us define $g = -a_{\Omega^*}|\nabla u_{\Omega^*}|^2 + d_{\Omega_T}$. A classical change of variable yields:

$$S(\Omega_\theta) = J((Id+\theta)(\Omega^*)) = \int_{\Omega^*} g \circ (Id+\theta)|\det(I+\nabla\theta)| \, dx = \int_{\Omega^*} g \circ (Id+\theta) \, dx,$$

where we used the fact that Ω_θ and Ω^* have the same measure. We use the following expansion

$$g \circ (Id+\theta)(x) = g(x) + \nabla g(x)\theta(x) + o(\theta),$$

for obtaining

$$S'(\Omega^*)(\theta) = \int_{\Omega^*} \nabla f(x)\theta \, dx.$$

Observing that $\nabla g\theta = \text{ div } (g\theta) - g \text{ div } \theta$ and that $\text{ div } \theta = 0$, we end up with the desired formula. □

The first order optimality condition of Theorem 3.5.2 evaluated at Ω^* gives:

$$\forall \theta \in \Theta_{ad},\ \ S'(\Omega^*)(\theta) = K_{\mathcal{O}},$$

where $K_{\mathcal{O}} \in \mathbb{R}$ is a constant value.

3.5.4 Conclusions

The existence problem discussed in this Chapter is still an open problem. Evaluating the optimality conditions inferred in the spaces $\mathcal{O}_V$, $\mathcal{W}_V$ has revealed difficult because it relies on the comparison of two terms: the 'compliance' one, depending on the gradient of the solution u_Ω and the 'distance' one, which is not defined in variational form. Note that using other notions of shape sensitivity (as for instance the topological derivative) does not help in escaping this issue. One way for addressing this problem could consists in writing the functional $J(\chi)$ (or equivalently $J(\Omega)$) as the energy functional associated to the solution of an Eikonal type equation. However, the analysis of optimality conditions seems suggest the existence of an homogenization region in the neighborhood of the boundary $\partial\Omega_T$. Some numerical tests that we performed for some particular choice of the target shape Ω_T seem reveal that in general the problem does not admits a classical solution for $\gamma > 0$.

Chapter 4

Shape reconstruction using elastic deformable surfaces

Contents

4.1 Introduction

In this chapter we will present an useful computational tool allowing to define a close surface model from a source unorganized model. This is a joint work with Maya de Buhan.

In many applications, a shape needs to be defined as the portion of the space that is bounded by an orientable surface. Such a description partitions the ambient space into two unambiguous subdomains, corresponding to the exterior and interior parts of the shape under consideration. Many geometry processing tools and mathematical models -as tetrahedral mesh generation, finite elements computations, shape representation, implicit modeling - requires such an unambiguous definition of the geometry of the underlying shape. Unfortunately, in many situations the geometry underlying the discrete data is unorganized or too complex to be described in terms of a connected subdomain of the ambient space. In such a situation, shape reconstruction and topological simplification tools may come into play. Shape reconstruction methods have been extensively investigated in the context of interface evolution via

level set methods (see [28] and references therein) or via deformable surfaces [39, 78]. In our specific context, we focus on the reconstruction of a connected shape starting from the discrete data of a source triangulation. Such a triangulation could present one or more undesirable peculiarities:

- *invalid*, showing for example overlapping entities;
- *not closed*, showing holes of arbitrary shape and dimension;
- *topologically dirty*, showing a huge number of connected components, which can be intrinsic of the object or issued from errors during the acquisition process (this is the case, for instance, of data coming from segmentation of noisy images).

Our specific task is to define a closed orientable surface model of the source triangulation so that it can be a guide for generating a tetrahedral mesh of the $3d$ bounded domain delimited by the generated surface. Most available geometry processing tools preserve the topological structure of the source object. In the present context we want both a surface reconstruction tool and a topology simplification tool. The latter can be useful to deal with shape representation, parametrization, retrieval. The specific application we have in mind will be described in Chapter 7. In order to achieve our goal we propose an original warping algorithm based on mesh deformation techniques. A closed surface mesh model is iteratively deformed, producing a sequence of surfaces (and corresponding triangulations) which are 'closer and closer' to the source triangulation, in the sense of the Hausdorff distance. Since the shape topology is kept unchanged during deformation, the algorithm generates a closed triangulation warping the source data. The proposed method has been implemented in two Finite Element settings. At first, in Section 4.3 we make use of the equations of three-dimensional elasticity acting on a thick template domain and at the end we extract the surface of interest. Then, in Section 4.4 we propose an elastic shell model which is able to deform an input surface. Starting from an input triangulation, the latter method generates three-dimensional finite elements taking the bi-dimensional surface and the normal vector field at each point of the surface as missing dimension. Let us now present the general framework of the method.

4.2 Presentation of the method

Let Ω_0 be a template shape with boundary $\partial\Omega_0 = \Gamma_0 \cup \Gamma_e$ composed of two disjoint closed surfaces Γ_0 and Γ_e. Let $\mathcal{T}_S$ be the source model which is known only in terms of the discrete data of a triangulation. Suppose that Ω_0 encircles $\mathcal{T}_S$. We made this assumption having in mind a precise application which will be exposed in Chapter 7; however the method can be easily generalized to the case of two model intersecting or to the case in which $\mathcal{T}_S$ encircles Ω_0. Our goal is to define a closed surface model of $\mathcal{T}_S$, which we achieve numerically by deforming a computational mesh of Γ_0 onto $\mathcal{T}_S$. The initial shape Ω_0 is iteratively 'deflated' in order to produce a sequence of

shapes $(\Omega_k)_{k=0,\dots}$ (and corresponding meshes $(\mathcal{T}_k)_{k=0,\dots}$) which get closer and closer to $\mathcal{T}_S$. Each step of the procedure is composed of two phases. First, the shape Ω_k is advected; the deformation is driven by the elasticity equations (1.5.3) with boundary conditions which we precise in the following. Then, potential intersections between the advancing model and the source model are checked. If a vertex x of the template shape has intercepted an element of $\mathcal{T}_S$ of during the deformation, then its coordinates are updated so that to force the point x to stay clamped at the source triangulation.

At each iteration the template domain is updated according to:

$$\Omega_{k+1} = (I + v_k)(\Omega_k). \tag{4.2.1}$$

The displacement field v_k featured in 4.2.1 is defined by:

$$\forall x \in \Omega_k,\ v_k(x) = \begin{cases} u_k(x) & \text{if } u_k(x) \cap \mathcal{T}_S = \emptyset \\ y - x & \text{if } u_k(x) \cap \mathcal{T}_S = \{y\} \end{cases} \tag{4.2.2}$$

where u_k is the solution of (1.5.3). Let Γ_k be the transformed of Γ_0 at step k. Let Γ_k^D and $\Gamma_k^N := \partial\Omega_k \setminus \Gamma_k^D$ be respectively the Dirichlet (the clamped subset) and Neumann (the subset submitted to surface loads) boundaries of the domain Ω_k. At each step k, Γ_k^D and Γ_k^N are allowed to vary according to the ray/surface intersection test featured in (4.2.2): if an intersection point y is found, then x is clamped at this point, so that $\Gamma_{k+1}^D = \Gamma_k^D \cup \{y\}$ and $\Gamma_{k+1}^N = \Gamma_k^N \setminus \{y\}$. Doing so, starting from Ω_0, which is clamped at $\Gamma_0^D = \omega$ and deflated on $\Gamma_0^N = \Gamma_0 \setminus \Gamma_0^D$, at each iteration the shape Ω_k are advected according to Equation (1.5.3) until they intercept the triangulation $\mathcal{T}_S$. Whenever a contact between the advecting shape and the source triangulation occurs, the first is clamped and forced not to cross the boundary of the latter. Doing so, at each step k the intersection between the vector $u_k(x)$ and the triangulation $\mathcal{T}_S$ is checked for all x vertex of Γ_k^D. In case of multiple intersection points, the point y closest to x is retained. By construction, the advecting sequence of meshes $(\mathcal{T}_k)_{k=0,\dots}$ gets closer and closer to the source mesh $\mathcal{T}_S$ guaranteeing the strict inequality:

$$d_H(\Gamma_{k+1}, \mathcal{T}_S) < d_H(\Gamma_k, \mathcal{T}_S),$$

where $d_H(\cdot,\cdot)$ is the Hausdorff distance between the two triangulations. Indeed, for k sufficiently large the advecting surface Γ_k defines a closed boundary that warps the source triangulation $\mathcal{T}_S$. From the numerical point of view we devised two types of Finite elements fro modeling the problem.

4.3 Model using tetrahedral elements

For the purpose of using Equations (1.5.3) we need to encode the template shape Ω as a solid medium, even if at the end of the story we will interested only on its inner surface Γ_k. Consider an initial closed template surface Γ_0. Consider the surface

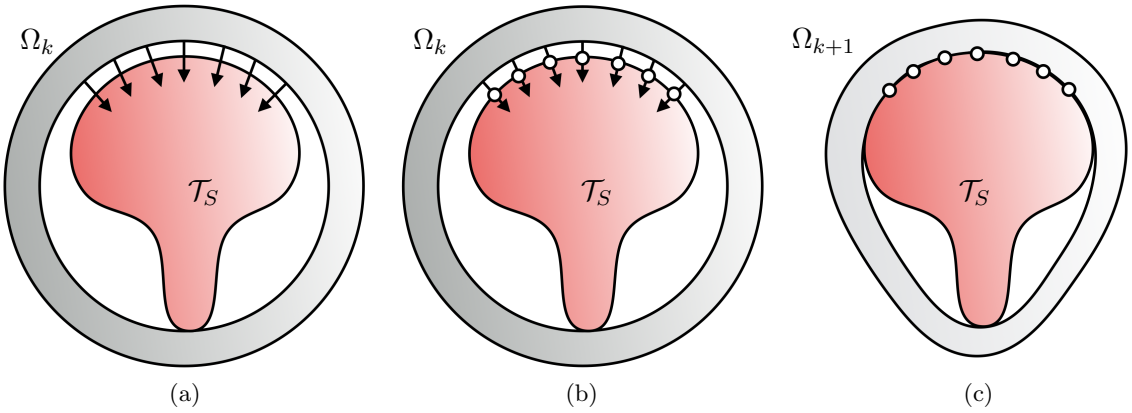


Figure 1: Schematic description of the warping procedure. (a) Computation of the displacement field u_k(black arrows) over the template shape Ω_k (grey color) encircling the source model (red color). (b) Retrieval of intersection points between the displacement vector field and the source model (white points). (c) Updating of the shape according to the displacement v_k.

obtained by thickening Γ_0 in normal direction:

$$\Gamma_e = (I + an)(\Gamma_0),$$

where $a \in \mathbb{R}^+$ is a 'small' parameter standing for the thickness of the solid and n stands for the unit normal vector to Γ_0, pointing outward. Consider the shape Ω_0 defined a the domain of $\mathbb{R}^3$ bounded by the two surfaces Γ_0, Γ_e. Assume that all the shapes $(\Omega_k)_{k=0,\dots}$ are clamped at a subset ω of Γ_0. The displacement field u_k at each iteration is computed as the solution of linear elastic problem 1.5.3 with the following boundary conditions:

$$\begin{aligned} \sigma(u_k) \cdot n_k &= p \qquad p \in \mathbb{R}, p > 0, \text{ on } \Gamma_k^D \cap \Gamma_k, \\ \sigma(u_k) \cdot n_k &= 0 \qquad \text{on } \Gamma_k^N \cap \partial\left(\Omega_k \setminus \Gamma_k\right), \\ u_k &= 0 \qquad \text{on } \Gamma_k^D, \end{aligned} \tag{4.3.1}$$

where n_k stands for the unit normal vector to $\partial\Omega_k$, pointing outward to Ω_k.

4.3.1 Numerical issues and examples

As far as the numerical setting is concerned, the template shape Ω_0 is discretized as a computational mesh $\mathcal{T}_0$ (filled with tetrahedra), with boundary composed of two surface triangulations $\mathcal{S}_0$ and $\mathcal{S}_e$ (resp. outer and inner boundary). The inner triangulation $\mathcal{S}_0$ supplies a simplicial mesh of the boundary Γ_0 of Ω_0. The source model is known only in terms of the discrete data of a triangulation $\mathcal{T}_S$. In a preprocessing step, the two models are aligned in oder to guarantee that Ω_0 encircles $\mathcal{T}_S$ and that

there exists a common subset ω between Γ_0 and $\mathcal{T}_S$. This stage corresponds to a rigid deformation and scaling of the two meshes. Starting from the template mesh $\mathcal{T}_0$, we perform an iterative algorithm in order to get a sequence $(\mathcal{T}_k,\ \mathcal{S}_k)$ of meshes with decreasing values of $d_H(\mathcal{S}_k, \mathcal{T}_S)$. For the sake of simplicity, the procedure ends when a maximal number of iterations k_{max} is reached. The general procedure relies on the following steps.

Start from an initial shape Ω_0,
for $k = 0, \cdots, k_{max}$

1. compute the state u_{Ω_k} of the considered PDE system on Ω_k by FEM discretization;
2. for each vertex of Γ_k check intersections between $u_{\Omega_k}(x)$ and $\mathcal{T}_S$;
3. compute the discrete vector field v_{Ω_k} featured in (4.2.2);
4. updated the discrete boundaries corresponding to Γ_k^D and Γ_k^N;
5. advect the shape Ω_k according to the displacement v_{Ω_k}, so as to get

$$\Omega_{k+1} = (I + v_k)(\Omega_k).$$

At each iteration the solution u_k of the elastic system (1.5.3) related to the domain Ω_k is computed by the Finite Elements Method.

Remark 4.3.1. *The numerical implementation of the proposed strategy requires at each iteration the computation of the Finite Element solution of an elastic problem* (1.5.3)*. Note that, even if at each iteration both terms in 1.5.3 are updated, the mesh connectivity does not change (since the shapes are not remeshed), allowing an efficient reassembly of the elastic system.*

Remark 4.3.2. *Concerning the stopping criterion, one can devise the following alternative strategy. Given a fixed integer N, the procedure ends whenever the set Γ_k^N is empty (all the points in Γ_k^N have reached $\mathcal{T}_S$) or if N iterations of the process occur without registering a new ray/triangle intersection. The latter condition deals with the potential presence of large holes in $\mathcal{T}_S$. Alternatively, one could prescribe a fixed percentage of elements of the the triangulation $\mathcal{T}_0$ which have to intercept the source model.*

Computation of the ray/mesh intersection

From a numerical point of view, the intersection test between a 3d segment and a triangular mesh corresponds to verify the intersection ray/triangle for *all* triangles of the triangulation. Moreover, since the target shape is moving during the process,

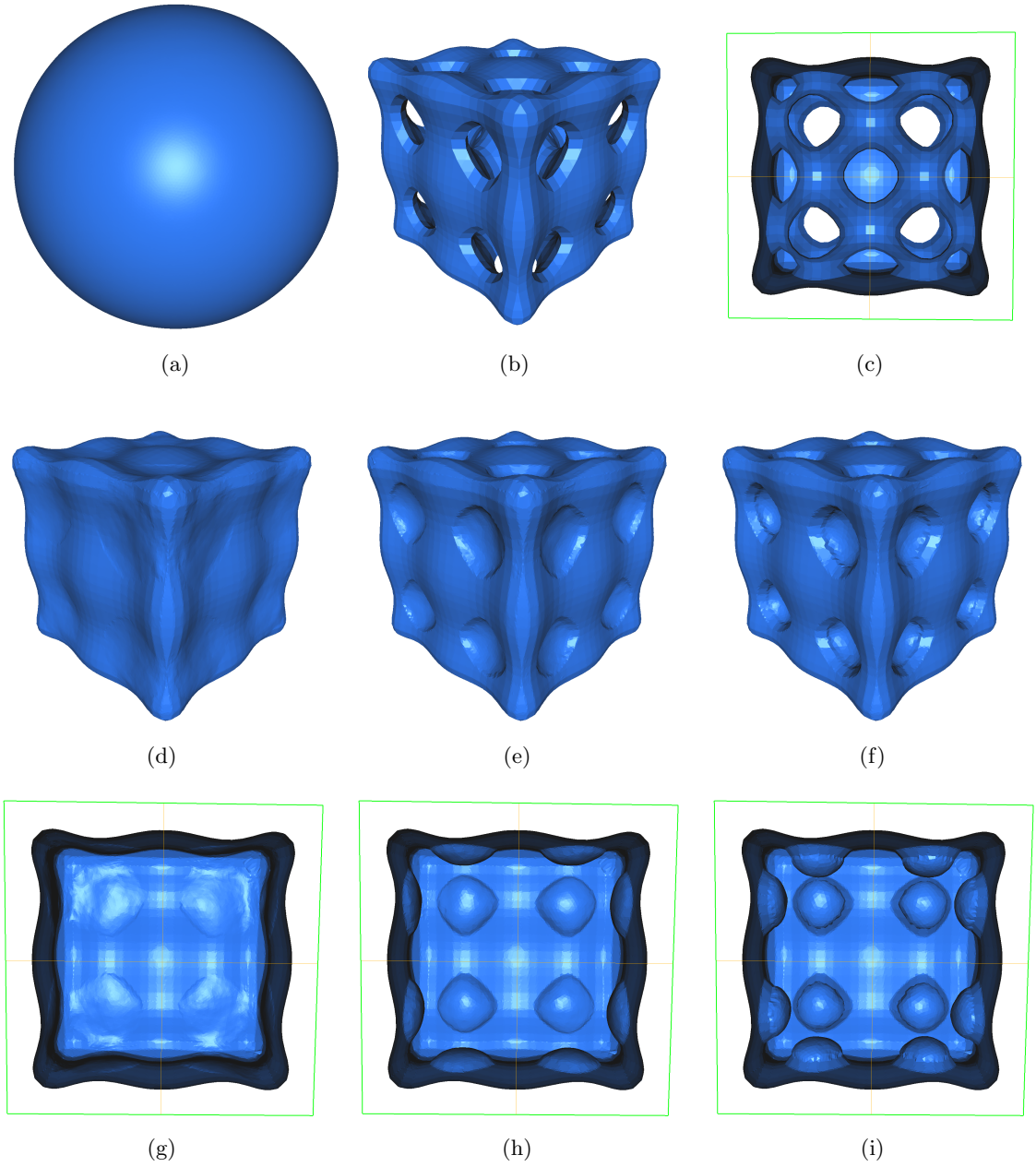
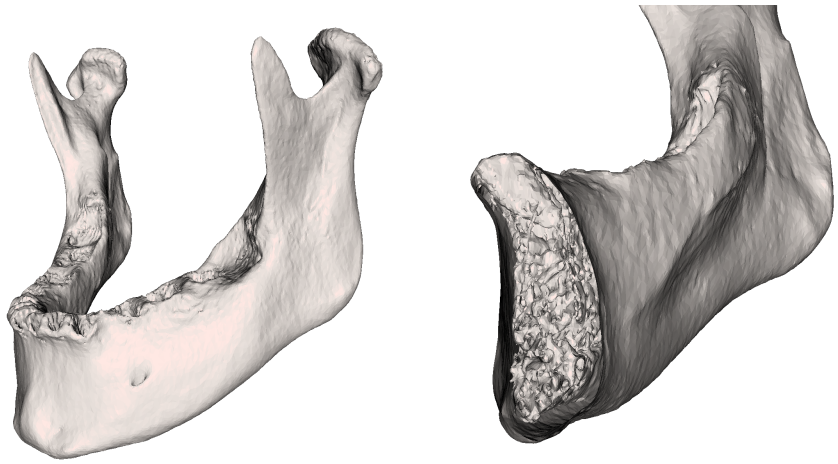
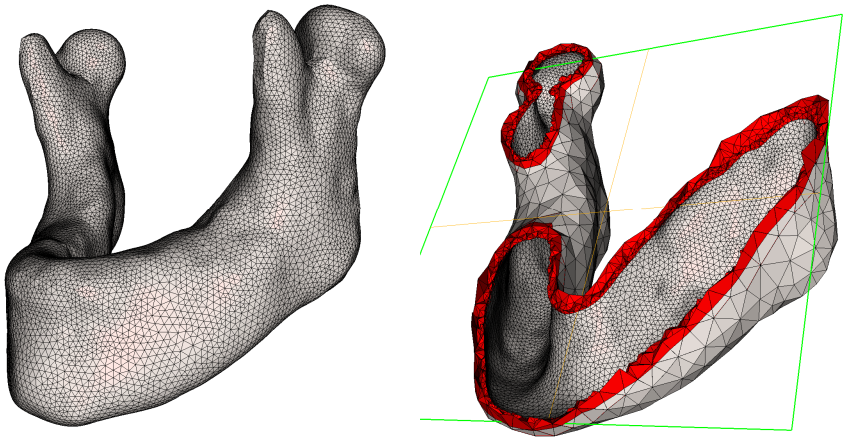


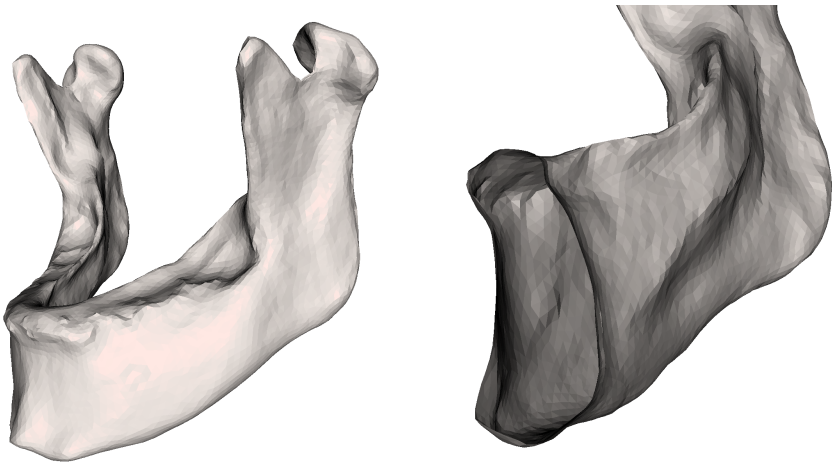
Figure 2: Topological simplification of a $3d$ model. (a) Computational mesh of the template shape. (b)-(c): Source triangulation. Warped model after $k = 10$ iterations (d)-(g); $k = 20$ iterations (e)-(h); $k = 80$ iterations (f)-(i).



(a)



(b)



(c)

Figure 3: (a) Mandible mesh model issued from segmentation of medical images. (b) Computational mesh of the template shape. (c) Closed surface model of the mandible generated after $k = 120$ iterations of the procedure.

we need to run the intersection test at each iteration of the process. Doing this computation in a naive way leads to a quadratic complexity, which is useless in practice. Our implementation takes advantage of a bucket sort algorithm in order to reduce significantly the number of intersection evaluations. The general idea is to endow the ambient space with a uniform grid structure (bucket structure) of low computational cost allowing to associate elements by means of their spatial vicinity. In order to do so, the joint bounding box of both source and target triangulations is partitioned using a uniform grid of non-overlapping cubic bricks of equal size. Each brick is indexed by a key integer depending only on the coordinates of its center. The triangles of the fixed source mesh are listed according to the buckets cells they intersect. When it comes to advect a point x in a prescribed direction u, a straightforward computation of the key allows to quickly retrieve the spatial neighborhood of the vector $u(x) - x$. Then the intersection test is performed only for the few triangles selected by this process.

Numerical examples

First, consider the example depicted in Figure 2. Both template and source model are embedded in a computational domain of dimensions $[0, 1]^3$. The template mesh $\mathcal{T}_0$ is a thick spherical template with about 20 000 tetrahedra. The source triangulation has about 15 000 triangles. The parameters we used for assembling the elastic system are

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)},$$

where $E = 10000\ kPa$ is the Young modulus and $\nu = 0.1$ is the Poisson coefficient, which corresponds to a very soft and compressible material. The pressure is $p = 200\ kPa$. Figure 2 depicts three possible results of the procedure for the following values of k_{max}: $k_{max} = 10$, $k_{max} = 20$, $k_{max} = 80$. Note how well the triangulation $\mathcal{T}_{k_{max}}$ approach source model, whose elements are still visible on $\mathcal{T}_{k_{max}}$. The computation takes about 20 minutes on a standard laptop computer.

Next, consider the example depicted in Figure 3. The source triangulation is an anatomical shape coming from segmentation of medical images. The triangulation is composed of several holes located on the outer envelope and filled at its interior with a complex network of triangles corresponding to the spongious bone (Figure 3-(a)). The source triangulation has about 60 000 triangles. The template shape is a thick domain which envelops the source triangulation (Figure 3-(b)) which has about 11 000 tetrahedra. The elastic parameters are the same of the previous example. The pressure is $p = 50\ kPa$. At the end of the procedure the parameter k_{max} equals 120, generating an high-quality closed model of the mandible (Figure 3-(c)). The computation takes about 25 minutes on a standard laptop computer. Some anatomical details of the source model have been lost (the mental foramen, the zones of insertion of teeth). Note that this issue can be overcame by simply adapting the size of elements of the template shape according to those of the source triangulation.

Remark 4.3.3. *The computational mesh of the template shape is not uniform. The inner triangulation is much more dense than the outer one. A control of the gradation parameter during the generation of the mesh allow the generation of high-quality tetrahedral elements, needed for the fast convergence of the Finite Element computation.*

Remark 4.3.4. *The computational cost of each Finite element computation increases remarkably with the number of elements of the support mesh. For improving the efficiency of the overall algorithm, one can devise a mesh adaptation scheme which, starting from a coarse computational mesh, refines the mesh as closer as the source model is approached.*

4.4 Model using shell elements

For avoiding the need of of generating a thick template (which we use only for the purpose of using $3d$ elasticity equations) in a preprocessing stage, we devised an alternative implementation of the method described in Section 4.2. This implementation relies on the shell theory. A shell structure is a three-dimensional solid structure in which one dimension is much smaller than the other two dimensions. We will now introduce some basic definitions needed for the mathematical description of shells. We follow the monography [20].

4.4.1 Shell elements

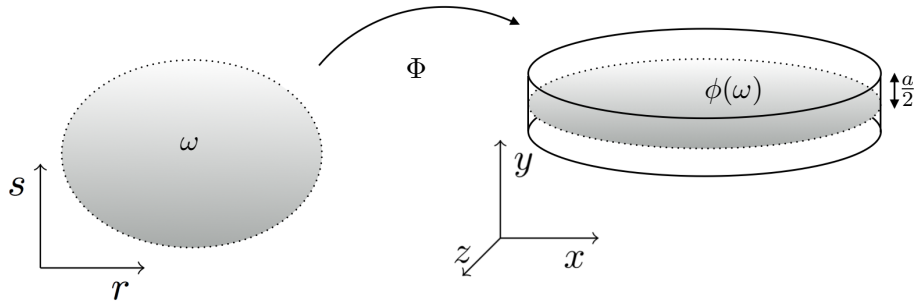


Figure 4: Geometric description of a shell structure. A bidimensional domain ω is mapped into the shell mid surface $= \phi(\omega)$. The enlarged domain Ω in (r, s, t) coordinates is mapped into a solid medium $\Omega_e = \Phi(\Omega)$ through the map Φ.

From the geometrical point of view, a shell is a tridimensional solid medium described in terms of a 'midsurface' immersed in the ambient space $\mathbb{R}^3$, and a parameter accounting for the thickness of the medium in the vicinity of the midsurface. From a theoretical point of view, the midsurface is described in terms of a collection of

two-dimensional charts, i.e. smooth injective mappings from domains of $\mathbb{R}^2$ to $\mathbb{R}^3$. Without loss of generality, we can consider the case in which the midsurface is defined in terms of a sole chart. The general case is treated by considering separately each chart of the collection. Therefore, let $\omega \subset \mathbb{R}^2$ be an open bounded reference domain. Suppose to know a chart ϕ:

$$\begin{aligned}&\phi : \bar{\omega} \to \mathbb{R}^3\\ &(r,s) \to (\phi_1(r,s), \phi_2(r,s), \phi_3(r,s)),\end{aligned}$$

which maps the bidimensional domain $\bar{\omega}$ into the midsurface $S := \phi(\bar{\omega}) \subset \mathbb{R}^3$ of the shell. Suppose also that the vectors $\frac{\partial\phi(r,s)}{\partial r}$, $\frac{\partial\phi(r,s)}{\partial s}$ are linearly independent at each point x of the midsurface, so that they span the tangent plane. Then, the unit normal vector $n = n(x)$ at x is defined according to:

$$n = \frac{\frac{\partial\phi(r,s)}{\partial r} \times \frac{\partial\phi(r,s)}{\partial s}}{||\frac{\partial\phi(r,s)}{\partial r} \times \frac{\partial\phi(r,s)}{\partial s}||}.$$

Also, consider the three-dimensional domain:

$$\Omega_a = \left\{(r,s,t) \in \mathbb{R}^3 : (r,s) \in \omega, t \in \left(-\frac{a}{2}, \frac{a}{2}\right)\right\},$$

obtained by enlarging the domain ω lying in the (r,s) plane with a thickness $a \in \mathbb{R}^+$ representing the thin dimension of the structure. The shell geometry is described in terms of the map:

$$\begin{aligned}&\Phi : \Omega_a \to \mathbb{R}^3\\ &(r,s,t) \to \phi(r,s) + tn(r,s).\end{aligned}$$

From a numerical point of view, the shell geometry is described in terms of two data: (i) a surface triangulation $\mathcal{T}_\Omega \subset \mathbb{R}^3$ representing the discretization of the shell midsurface; (ii) the normal vector field at vertices and a scalar function a: $\Omega \to \mathbb{R}^+$ which describes the thickness along the normal direction.

One can devise two kinds of shell deformation models: the solid models, formulated in terms of the coordinates of the ambient space $\mathbb{R}^3$, or surface models, formulated in terms of curvilinear coordinates defined on the midsurface (see for instance the monographs [22, 82]). Here we rely on the first approach, using the linear elastic equations defined in Section 1.5. Note that the linear elasticity framework is well defined for the shell structure which is a tridimensional medium. Our sole aim is to devise appropriate Finite Elements which could exploit the peculiar structure of the solid at stake. In order to do so, we consider tridimensional shell finite elements formulation taken from [9]. Let us explain briefly the Finite Element setting.

Consider a surface triangulation $\mathcal{T}$. The elements of $\mathcal{T}$ will be referred as the shell mid triangles. Consider the P^1 Lagrange $2d$ linear functions $(h_i(r,s))_{i=0,1,2}$ defined

on a reference triangle T^* in the (r, s) reference coordinate system:

$$h_0(r,s) = r, \qquad h_1(r,s) = s, \qquad h_2(r,s) = 1 - r - s.$$

Consider a shell midtriangle T_k. Let $(x^{(i)}, y^{(i)}, z^{(i)})_{i=0,\dots,2}$ be the coordinates of nodes $(p^{(i)})_{i=0,\dots,2}$, located on T_k and n_i^x, n_i^y, n_i^z, $i = 0, \dots, 2$ be the components of unit normal vectors at nodes (see Figure 5). Let T_k^e the shell element constructed by enlarging the midtriangle T_k along the normal direction (Figure 5-right). The cartesian coordinates (x, y, z) of a point inside the element T_k^e are expressed in terms of the shape functions $(h_i(r,s))_{i=0,1,2}$ defined on the (r, s) plane and the unit normal vectors at nodal positions, according to:

$$x(r,s,t) = \sum_{i=0}^{2} h_i(r,s)\big(x^{(i)} + t\frac{a}{2}n_i^x\big), \tag{4.4.1}$$

$$y(r,s,t) = \sum_{i=0}^{2} h_i(r,s)\big(y^{(i)} + t\frac{a}{2}n_i^y\big), \tag{4.4.2}$$

$$z(r,s,t) = \sum_{i=0}^{2} h_i(r,s)\big(z^{(i)} + t\frac{a}{2}n_i^z\big). \tag{4.4.3}$$

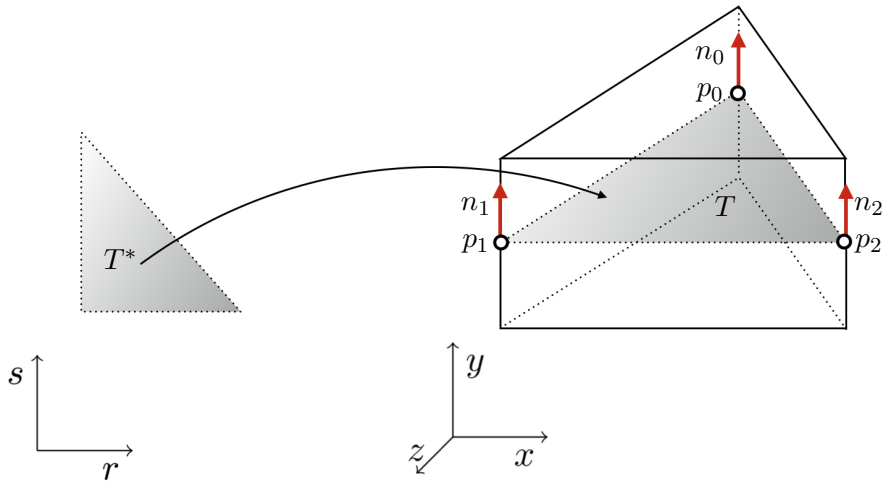


Figure 5: Mapping a reference triangle T^* in the (r, s) coordinates to a shell triangular element T^e, obtained by enlarging the mid triangle T (grey color) in the direction of the normal vectors (red color) defined at nodes (white circles).

Starting from this expression for points coordinates one can assemble the Finite Element problem in a standard way. See Appendix A for technical details about the strain/displacement interpolation. Note that in this context the Jacobian matrix of the mapping $(r, s, t) \to (x, y, z)$ is not constant in (r, s, t) even in the case of linear

basis functions, since the interpolation (4.4.1) does not depend linearly on the (r, s, t) coordinates. See Appendix A for an example of mesh deformation using this setting.

4.4.2 Numerical issues and examples

To end this chapter, we are going now to use the algorithm featured in Section 4.3 in the context of shell elements. In the present context, the algorithm is unchanged, except the fact that it takes as input a surface Γ_0 (and a corresponding triangulation $\mathcal{T}_0$) which represents the midsurface of the shell.

First, Figure 6 depicts the same test case of Example 4.3.1 when the template shape is discretized as a $3d$ spherical triangulation and the Finite Elements system is implemented using shell elements. The template shape has about 13 000 triangles. The elastic parameters we used are the same. The pressure is $p = 2000\ kPa$. The thickness value of the shell is fixed at $a = 0.3$. Note that a bigger value of the pressure (which we multiplied by 10 with respect to the previous example) allows to reach the source model with a much smaller number of iterations, without stretching or invalidating the advecting mesh. The computation takes about 10 minutes on a standard laptop computer.

Next, Figure 7 depicts the deformation of the same template Γ_0 of the previous example onto the source mesh model of a molecule having about 12 000 triangles. The source model is composed of several connected components, each one representing one atom of the molecule 7-(f). The elastic parameters and the thickness value are the same of the previous example. Thanks to our algorithm we are able to generate a closed surface model which is suitable for generating a tetrahedral mesh of the source object (Figure 7-(e)). The computation takes about 15 minutes on a standard laptop computer.

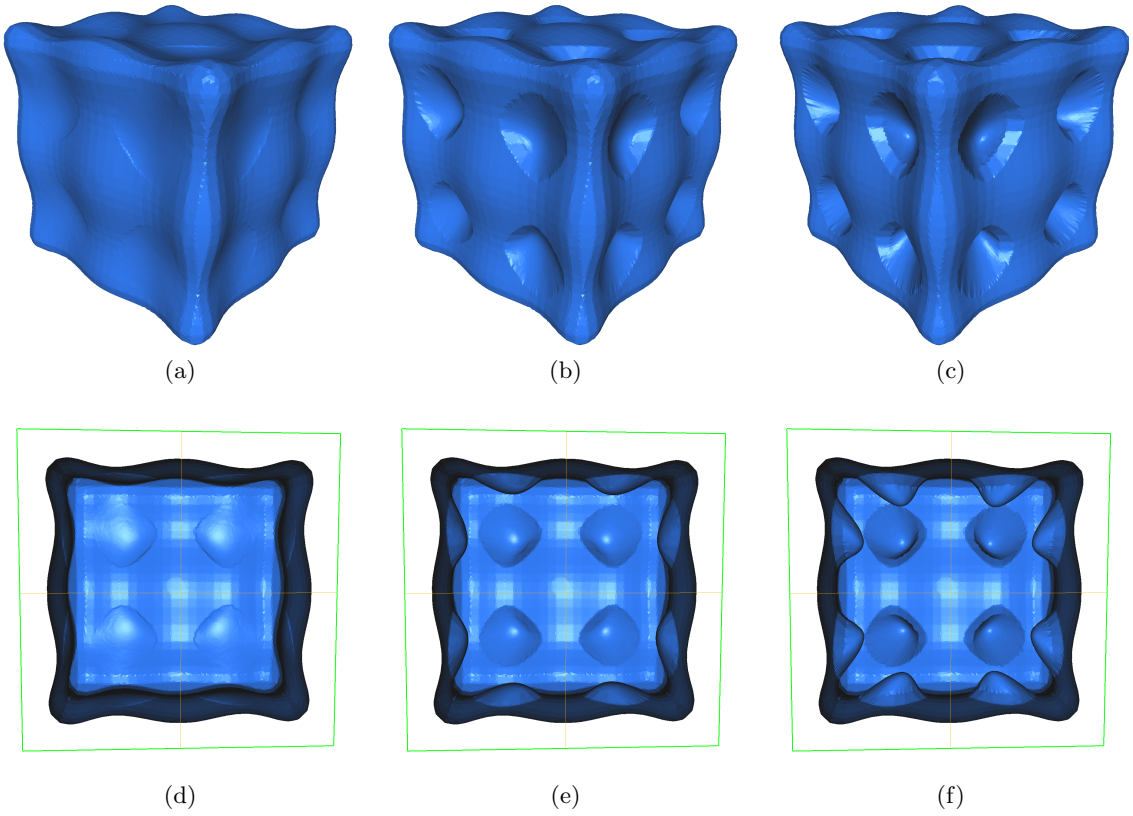
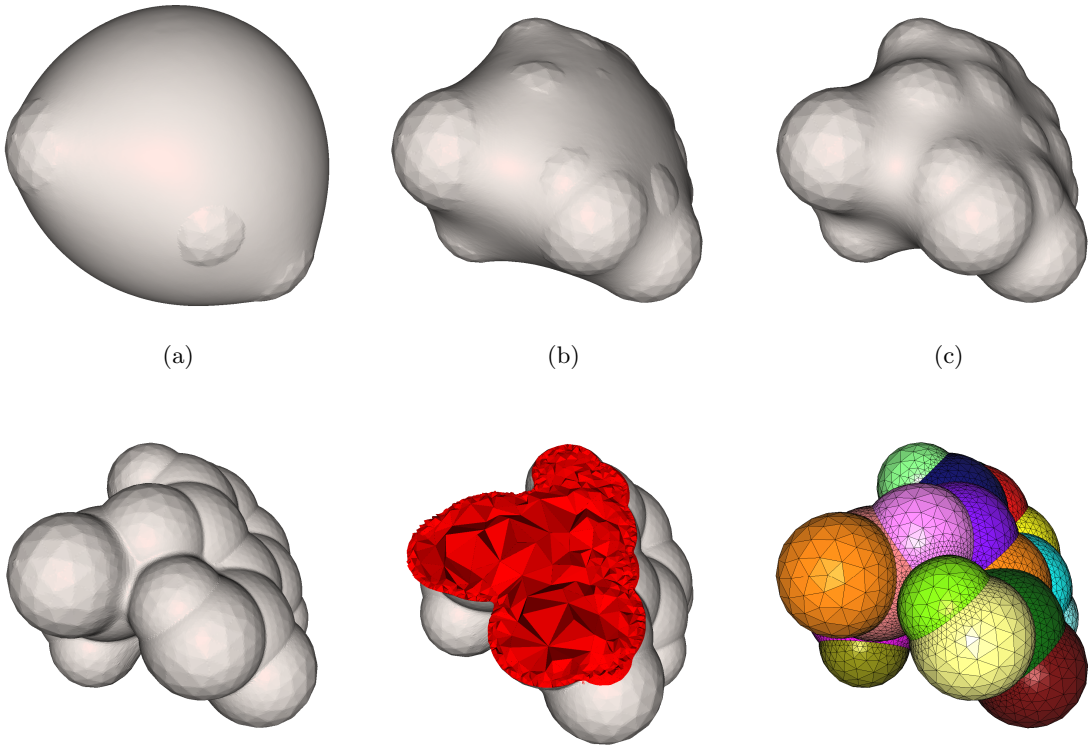


Figure 6: Test case of Example 4.3.1 when using shell elements. Warped model after (a)-(d)$k = 3$ iterations; (b)-(e) $k = 7$ iterations; (c)-(f) $k = 10$ iterations.



(a) (b) (c)

(d) (e) (f)

Figure 7: Warping of a molecule mesh model at iteration (a) $k = 3$; (b)$k = 7$; (c) $k = 10$; (d) $k = 49$. (e) Tetrahedral mesh of the subdomain bounded by our closed model. (f) Source model.

Part II

Digital Facial reconstruction

Chapter 5

Background and state of the art of facial reconstruction

Contents

5.1 Introduction

(Cranio)facial reconstruction methods (FR) aim at estimating the facial appearance of an individual from the sole datum of the underlying skull. Facial reconstruction practices are based on the assumption that there exists a mutual strong correlation between the bone structure and the soft tissues [62]. However the creation of the face from the skull is a procedure of *approximation*: from the observation of the cranium, one will not be able to recover a big amount of facial features (eyes color, hair, skin color, lips shape, ears). Moreover the facial likeness of a single individual changes considerably depending on factors like nutrition or aging. This face

flexibility is not reflected on the subjacent skull. While age and sex can be estimated from the observation of the skull, the BMI index can not be predict. A correct model for facial reconstruction should take into account these variations, eventually by proposing multiple reconstructions for one single skull (as proposed in [27, 79]). The final product of a facial reconstruction tool is then expected to reflect somewhere the intrinsic uncertainty related to this *ill-posedness* of the problem, no matter which method is employed (artistic, parametric, statistic, mechanical, etc.). From a mathematical point of view, this issue leads to at least two important difficulties: from one hand, how to correctly characterize the solution, which might be a continuum spectrum of all faces 'consistent' with a given skull rather than a single exemplar; from the other hand, how to rigorously assess the degree of accuracy of the result. Despite the intrinsic difficulty of the problem, the media are full of facial images that have been constructed on the basis of a single given skull. A fascinating survey of such cases can be found in the book [90]. The facial reconstruction problem arises in various application fields ranging from archeology to medicine. Facial reconstruction methods are currently employed in forensic medicine and in archeology. In forensic science, facial reconstruction comes in the process of identification of deceased people when all the usual methods of identification have failed and the skeletal remain is the sole datum available for leading to a positive identification. In this context facial reconstruction might be considered as an enhancing tool for 'recognition', producing a short list of candidates from which the individual may be identified by DNA appraisal or other endorsed methods of identification [118]. In archeological investigations, facial reconstruction is employed with the purpose of identifying skeletal remains of famous people from the past. The work presented in this part of the manuscript is part of the ongoing multi-disciplinary project FaciLe *, grouping together maxillofacial surgeons, anthropologists, computer scientists and mathematicians from Sorbonne Universités.

5.1.1 A brief history of facial reconstruction

Facial reconstruction has a fascinating history dating from the early dawn of civilization and having beautiful links with ancestor worship and burial practices. In 1930s a collection of plastered human skulls (7000 b.c.) was found under the floor of a Neolithic house in Jericho. Faces were sculpted over the skulls in plaster and shells were put in the ocular cavity to mimic eyes and to enhance lifelike appearance. In this context the symbolic meaning is preponderant over anatomical correctness; for example the lower jaw was systematically separated and removed from the rest of the cranium. However, the Jericho faces can be considered as the first attempt to facial reconstruction. Archeological founds record in different cultures the tradition of manufacturing death masks serving as portrait sculptures. Depending on period and culture, the death masks were built either by taking a wax cast from the face or by sculpting the anatomy directly from the bone structure. The work done by the Italian anatomists Ercole Lelli (1702-66) in Bologna, Gaetano Giulio Zummo (1656-1701) in Florence fall onto the latter category. Lelli realized a series of extremely accurate

*http://www.sorbonne-universites.fr/actions/recherche/chaires-thematiques/facile.html

Figure 1: Jericho plastered skulls (7000 b.c.) housed at British Museum of London (left) and at Jordan Archaeological Museum of Amman (right). Images reprinted from [90].

anatomical waxes by positioning wax copies of muscles and soft tissues over the bone armature with the purpose of studying anatomy and teaching medicine students. Wax anatomical modeling was already practiced in the Renaissance period by artists of the range of Michelangelo and Andrea del Verrocchio but it was in the seventeenth century that this discipline turned into the art known as of *plastica anatomica*. These anatomists had the merit to understand the correlation between the bone and the upon musculature. However none of the described attempts were made with the purpose of forensic identification. The first forensic facial reconstruction is credited to the German anatomist Wilhelm His (1831-1904) and the sculptor Karl Seffner (1861-1932) who worked together for producing the famous reconstruction of the bust of Johann Sebastian Bach from a copy of the skeletal remains. The procedure of His was based on the collection of measurements of facial soft tissue depths, which he measured on a small set of 28 cadavers. In the same period the anatomist Hermann Welcker (1822-1897) was able to conjecture the non authenticity of the skull of the German poet Schiller, which has been proved in 2008 by DNA comparison, and to verify the authenticity of the skulls of numerous historical individuals (Dante Alighieri among them [114]).

It was at the beginning of the twentieth century that the reconstruction of the face from the skull became a part of forensic science. A famous case of such an employment is recorded in New York in 1916, when the facial reconstruction of the disappeared men Domenico la Rosa led to a positive identification of the skeletal rest.

The interest in facial reconstruction increasing, two main reference schools were developed, namely the *Russian School* and the *American School*. The so-called Russian method is due to the Russian anthropologist Mikhail Gerasimov (1907-1970). The keystone of Gerasimov's approach was the assumption that the shape of facial muscles depends strongly on the underlying bone structure [47, 46]. Then both deep

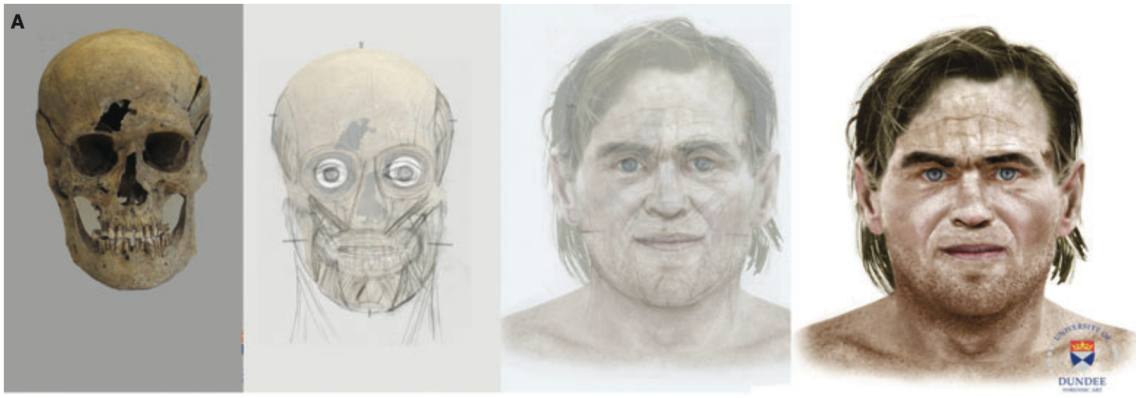


Figure 2: $2d$ manual method, reprinted from [119].

anatomical knowledge of the facial muscles variability and artistic skills were needed for practitioners of the Russian method. The Russian method does not make use of soft tissue depths tables. According to Gerasimov's records, the Russian method led to the successful identification of 150 forensic cases and to several historical reconstruction. A famous one is the reconstruction of the bust of Ivan the Terrible. Contemporary and in some sense opposite to the Russian school, the so-called American Method was pioneered by the anthropologist Wilton Krogman [67]. Instead of requiring the reproduction of the anatomical details, the American method is based on tables of the measurement of soft tissue thickness. The tissue depth tables are ranged according with age, sex and ethnicity and used for estimating the position of facial landmarks. This method basically consist of three steps:

- equip (a replica of) the raw skull with a sparse set of anatomical landmarks;
- apply an average soft tissue thickness to each skull landmark in order to estimate a corresponding landmark on the face;
- draw up or sculpt a face fitting the estimated landmarks;
- finalize the reconstruction adding artistic features.

Many computer based methods are based on the American method.

In the 80s in England a new school was developed, leading to the so-called Manchester method). This method combines feature from both American and Russian Method and was developed by Richard Neave and Carolin Wilkinson. Traditional methods based on manual procedures include also $2d$ methods based on portraits [102, 57].

5.1.2 Difficulties with traditional methods

The results obtained from forensic art are often quite plausible, as the medical artists may take anatomical, historical, archaeological or other type of expertise into account, giving the observer a feeling of coherence. However, the manual methods encounter some criticisms, due to the lack of methodological standardization and the subjectivity of the reconstruction. First, manual reconstructions require a profound anatomical knowledge of the human face and remain difficult. Moreover, one single reconstruction requires several days of work of a well-experienced forensic artist, making impracticable the realization of multiples instances and feature variations. The most important problem with manual methods is the subjectivity of the reconstruction. A famous examples reported in [57] consisted in asking to nine forensic artist to realize the facial reconstruction of some victims of the Green River Serial Murderer (1982) from skeletal remains. In this experiment 24 facial reconstruction methods has been compared. The reconstructions made by the different artists showed great variance, even if in some cases they were quite accurate. For employment of facial reconstruction methods in criminal cases it is crucial to establish a validation protocol and a range of reliability. Such a protocol has to be established in the basis of quantitative analysis and cannot take into account the subjectivity of the artist.

In order to alleviate these difficulties, with the booming of computer science and medical imaging techniques, several computer graphics tools have been developed with the purpose of assisting the facial reconstruction. These animation softwares mimic the methodology of manual methods, allowing the expert to vary some modeling parameter and combining the human expertise with the flexibility of the software. See [79, 110] for examples of such a procedure. However, computer assisted non-automated methods do not eliminate the attribute of subjectivity in the reconstruction. During the last 30 years an important deal of work has been devoted to the conception of objective fully-automated methods. A survey and review of 25 facial reconstruction can be found in [25].

5.1.3 Digital $3d$ methods

Digital methods shows major advantages as the absence of subjectivity, the rapidity of the processus, and the versatility of the model, allowing to explore several morphological hypothesis or verify the sensitivity of the model with respect to the variation of some parameters. Most actual approaches associates statistical models and database.

5.1.4 General framework of digital FR methods

According to [25], the common framework of an automated facial reconstruction software reposes on the following phases: 1) database acquisition; 2) anthropological examination and digitization of the unknown skull; 3) target skull representation; 4)

Figure 3: $3d$ manual method, reprinted from [119].

assessment of the craniofacial model; 5) model to skull registration; 6) texturing and visualization; 7) validation.

The most salient differences between all the current methods concerne the choice of representation for the unknown skull, the choice of the craniofacial model and the choice of the admissible geometric transformation for registering the craniofacial model to the unknown skull.

5.2 Database acquisition

Current numerical craniofacial reconstruction methods all repose on the a priori information contained in a database of coupled skull and face templates. The acquisition of both skull and face is beastly accomplished by head CT scans of living subjects, allowing a good visualization of hard tissues. Unfortunately the invasiveness of this technique causes serious legal and ethical problems, making troubling the constitution of a large database of healthy subjects. Due to this difficulty, several studies have proposed to exploit the relationship between soft and hard tissues by means of average soft tissue thickness measurements. Soft tissue depth tables are at the basis of all the traditional facial reconstruction methods and of several computer-aided methods. Originally acquired by needle punctures on deceased people, these measurements are currently obtained by CT scans [104], MRI, cone-beam CT scans [63], or ultrasound systems [36]. When using average soft tissues depth studies the reconstruction method could be based on the combination of a large database of face templates together with average soft tissues measurements (as in [25]). Facial templates can be acquired by non-invasive techniques such as stereophotogrammetry [55] making the constitution of a large database painless and easy. However there are issues with the use of soft tissue averages for predicting the facial morphology (see [68]). Moreover, the average values of soft tissue thickness are systematically measured on a sparse set (< 53) of (manually positioned) anatomical landmarks. Since the manual measurement is time consuming and requires expertise in correctly identifying the landmarks, it is actually infeasible to extend these measurements to a

dense distribution of points. Due to this limitation, several authors have claimed the importance of using a dense representation describing the skull and the face in terms of dense surfaces ([104, 84, 91]) and the correlation between soft and hard tissues in terms of the volume between the two boundary surfaces ([84, 91]). An alternative approach is proposed in [87], in which the authors use a face database of IRM images together with CT scans of dry skulls and a small collection of IRM of both skulls and faces.

For our study we used a small collection of 28 head CT scans of female healthy subjects aged between 20 and 40 years from the database [105]. CT images are acquired from subjects in a horizontal position. As a result, due to gravity effects, facial shapes extracted from CT images differ from the typical facial shape as viewed in a standing upright position. This horizontal-to-upright discrepancy might affect the recognition of the correct subject. Due to the difficulty of collecting full head scans of healthy subjects and the time consuming segmentation of the data, the existing CT scan databases are very limited. This lack of adequate data creates serious difficult for benchmarking existent methodologies, for which it is crucial to use a standard test data set for researchers for comparing the outcomes. We mention the database of Kustar et al. [68] composed of full head CT scans of healthy subjects in upright position.

5.3 Anthropological examination and digitization of the unknown skull

A basic step common to all existent FR methods is an anthropological examination of the unknown skull. This procedure consists in taking craniometric measurements in order to discriminate parameters like age, gender. When pelvic bones are unavailable the skull is considered the best indicator of sex [48, 116]. Also, age can be estimated from the cranium. The accuracy of those sex determinations varies considerably between different osteological elements, and also between different human populations [43]. After anthropological examination, a virtual replica of the dry skull is produced and encoded into the model according with the choice of the skull representation. Currently the most popular method for digitalizing the unknown skull is the acquisition by CT scans (producing gray-level stacks of 2d images) combined with semi-automatic segmentation (producing a 3d mesh model).

5.4 Target skull representation

The choice of the parametrization for the skull template depends strongly on the nature of the database. The human skull is characterized by a quite complex structure, showing very small details which are difficult to acquire, due to the limitations in resolution of the scan devices and the errors cumulated in the segmentation pro-

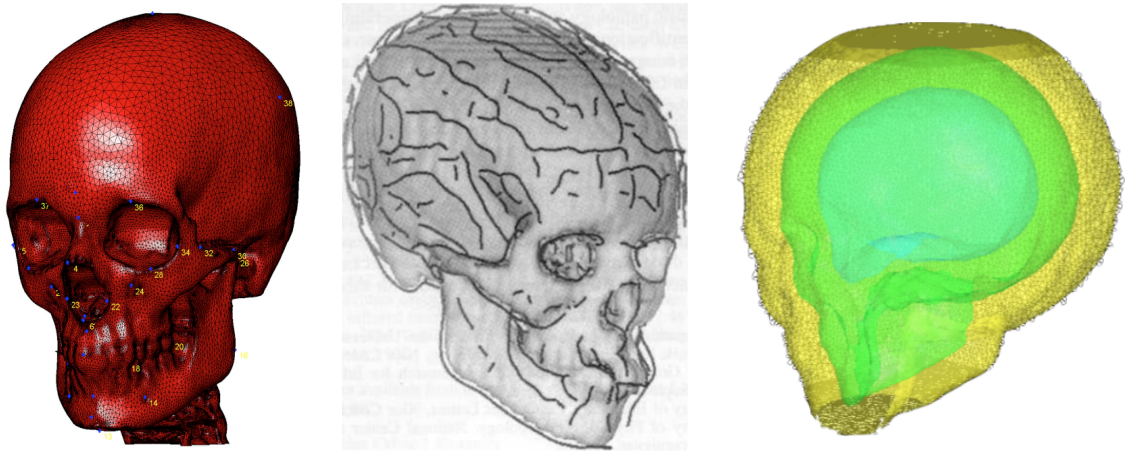


Figure 4: Three representation of the target skull. (a) Anatomical landmarks, reprinted from [103]. (b) Crest-lines, reprinted from [91]. (c) Level sets of the signed distance function.

cess, and to handle numerically, especially in the process of meshing or matching skull templates in the database. Due to these difficulties, several authors opt for describing the skull in terms of underlying anatomical or geometrical substructures. In [27] the skull is described in terms of 52 estimated landmarks obtained by a database of face templates and soft tissues depth measurements. In [91] authors use automatically detected continuous crest-lines. Most of the current methods are based on a dense representation of the skull. In [24, 109] the skull model is implicitly described by the euclidean signed distance function to its boundary. However in most methods the dense description of the skull is coupled with a sparse set of feature points (anatomical landmarks). The feature points drive the process of matching skull templates. See Chapter 6 for a discussion about the use of landmarks in anthropology.

5.5 Definition of the craniofacial model and registration to the unknown skull

The assessment of the craniofacial model refers to the following three elements: modeling of the craniofacial database; choice of a reference face template to be deformed on the unknown skull; choice of the admissible geometric transformation for deforming the reference face incorporating the soft tissue information coming from the database. After setting up the parameters of the model, the craniofacial template model has to be adapted -or registered,warped- onto the unknown skull. Most current methods use statistical models [13, 27, 106, 37, 100] combined with non rigid deformation driven by feature points. The modeling of the craniofacial database refers to the representation of the skull and face templates (sparse landmarks, crest lines, level set functions, continuous surfaces) and the face/underlying skull relation

(soft tissue measurements, muscles, volumes). Regarding the storage of soft tissue information we can distinguish two main approaches: methods based on soft tissue depth tables and volumetric methods. Basically methods based on soft tissue depth tables are based on two stages. First, a set of corresponding landmarks is selected on both osseous and facial couples of a large database. The collection of soft tissue tables, initially performed on cadavers, is currently accomplished thanks to medical devices as ultrasons, X-ray based CT or CBCT. Second, a sparse set of parameters is extracted by measuring the distance between each skull landmark and the homologous face landmark. The soft tissue parameters are then classified according to sex, age, ethnicity and IBM for inferring average values of the soft tissue thickness inside a population. Albeit the information provided by this kind of approach is anatomically meaningful, because the landmarks are reproductibles by a well trained practicer, there are some issues with the use of soft tissue tables

5.6 Texturing and visualization

In a last step, in order to achieve a better life-like appearance, secondary features as skin color and hairiness should be applied to the reconstructed facial shape. This final step is accomplished by mapping a texture onto the geometric model of the reconstructed face. This stage is not without issues: albeit the recognition and identification of an individual is hard when looking at the sole facial geometry, adding a specific individual texture can false the recognition. When no additional information about the unidentified individual is available, probably the best choice is to add a generic texture, generated for instance by averaging specific textures.

5.7 Validation

The validation of a model for facial reconstruction is of tremendous importance for the purpose of legitimating its use during a criminal investigation. The right question to ask is the following: assuming that skull and face are deeply correlated, is the correlation enough to ensure the predictability of the face from the skull? As pointed out in [118], most of digital FR methods have not been tested for accuracy and reliability. A quantitative validation step as well as a comparison between existent methods is of crucial importance. The debate about reliability of facial reconstruction methods is controversial and there is no unanimous opinion within the scientific community. A first attempt to address this problem is a leave-one-out scheme. In practice one individual is removed from the database and the method is employed to reconstruct its face given the sole skull. In a second stage the predicted shape is compared with the available original face. The discrepancy between the two shapes can be evaluated mathematically by computing distances between them (Euclidean distance, Hausdorff distance, Gromov-Hausdorff distance, Wasserstein distance). However, since the final purpose of the method is a positive identification, a recognition test can be also

employed for revealing the power of prediction of the method [26]. A recognition test consist in showing the predicted face together with a sample of faces which contains the original face. Then the human volunteer indicates the face (or faces) which is (are) closer to the predicted one. The positive outcome will then correspond to the identification of the original face among the sample.

Chapter 6

Data processing

Contents

6.1 Introduction

All current Facial Reconstruction methods repose on the observation of a head database containing information regarding soft tissue variability. An adequate database for Facial reconstruction must contain couples of faces and skulls coming from an healthy individual together with anthropological parameters like age, sex, BMI. For achieving a good visualization of both bone and soft tissue -and facilitate the process of data segmentation- we use medical images coming from X-ray CT scans. Due to the invasiveness of the technique, our data comes from individuals who underwent a clinical examination. The denomination 'healthy individual' in this context means that the patient didn't show morphological anomalies in correspondence of the zones of interest.

The purpose of this chapter is to describe the data processing tools we needed for the establishment of our database. The establishment of the database has been done in collaboration with Lydie Uro and the others partners of the project FaciLe. Section 6.2 describes the acquisition of medical images we needed to build our 3d shape database. Since we are interested in studying the soft tissue informations, we were constrained to work with medical data coming from living and healthy subjects.

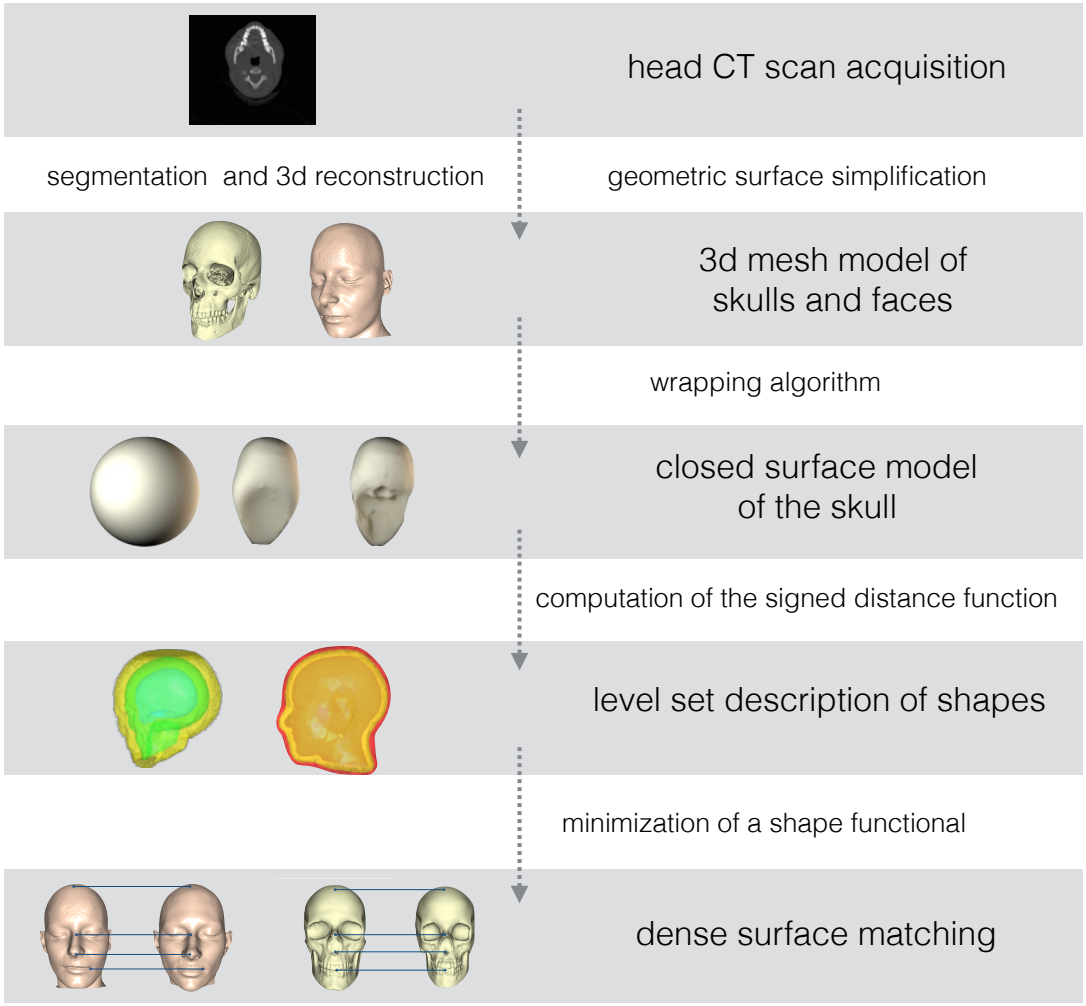


Figure 1: Stages of the data acquisition process.

Section 6.3 describes the procedure of segmentation of medical images for extracting the anatomical structures of interest (bone and facial envelops). Section 6.4 describes the generation of the geometrical model ($3d$ high-quality meshes) from the segmented images. In Section 6.6 we discuss about a common choice in data analysis in the field of computational anatomy: the definition of sparse landmarks for encoding the information related to the data. We discuss advantages and drawbacks of such a choice. However, since our main concern is the establishment of a fully-automated method, we choose to not define landmarks for our analysis. Eventually Section 6.7 describes briefly how to take advantage of the shape morphing method introduced in Chapter 2 for inferring correspondences between shapes.

6.2 Acquisition of a CT scans head database

The data collection we used for our experiments is composed of full head CT scans of 26 healthy female aged between 20 and 40 years. The BMI index of the subjects is known but up to now not included in our study. The CT images used have been provided by the facial reconstruction project of University Paris Descartes [105]. The X-ray based acquisition protocol is described in [105]. An independent study carried out in the context of the project Facile by L. Uro and Y. Maday aims at segmenting some facial muscles which have considerable influence on facial shape, in terms of functional importance or volume. In the context of this study a database of masseter muscles has been generated from 36 partial CT scans coming from both the database [105] and a database of the Pitié Salpêtrière Oral and Maxillofacial Surgery department. Unfortunately this latter series is composed of partial scans which were excluded from the craniofacial database. Despite the CT scan is the best medical imaging technique for achieving a good visualization of both bone and soft tissues, the use of CT scans database for facial reconstruction is far from being unanimous. Some authors have raised up disadvantages to the utilization of CT scans arguing essentially on three aspects. First, the invasiveness of the technique (armful radiations) makes serious legal and ethical issues for the acquisition of CT scans out of the clinical scope. Conversely, most of the CT scans coming from clinical analyses have been acquired from patients presenting pathologies and morphological anomalies, so they have to be removed from the database for preventing the risk of bias. Thereby it is difficult to build up a huge database of full head coming from 'healthy' people. Second, most of the case the individuals are disposed in supine position; due to the gravity effect, their face shape is deformed with respect to the unstressed one. Especially in some zones of the face this deformation risks to bias the database. Third, CT scans often presents artifacts due to dental filling which compromises the quality of several slices and complicate the process of segmentation.

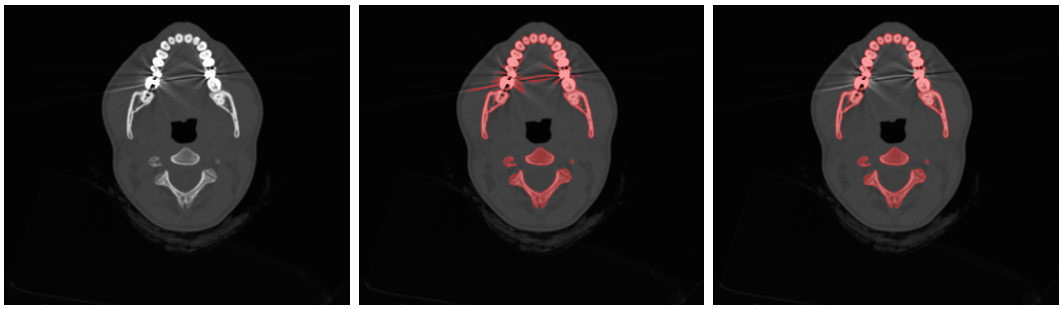


Figure 2: Left: reference gray-level image. Middle: pre-segmentation of the bone using a threshold technique. Right: segmentation of the bone after denoising and artifacts removal.

6.3 Segmentation of CT scans

The data segmentation consists in identifying the bone and soft tissues on the stacked 2d gray-level images. This procedure has been carried out in a semi-automated way. First, the CT slices are automatically pre-segmented using a multi-threshold technique. This step consists in partitioning the original images into subdomains which boundaries are identified by given intensity values. The bone and soft tissues threshold values we used are described in [105]. From the pre-segmentation step we obtain two sets of binary images, respectively for bone and soft tissues. By stacking these slices we can essentially detect the 3d structures. However the intensity-based segmentation is not enough to ensure a correct separation of the tissues of interest, due to the presence of noise on the data and artifacts occurred during the acquisition process. The binary images are then cleaned by removing the so-called islands (very small structures which contours are defined by only a few pixels). These structures can be external to the tissues of interest (noise added during the acquisition process) or very thin internal structures (small bones inside the cranium). This action helps in denoising the images. Moreover in most of the subjects we observed large artifacts on the images due to dental filling. These defects need to be manually removed on each affected slice (see Fig. 2). After the correct identification of the subdomains of interest, we proceed in separating the cranium from the vertebral column. This operation is also done manually by selecting contours on the slices interested by intersection between the two bone structures and labeling them. See Fig 3. Finally the column vertebral is excluded from the bone label because it has not influence on the facial appearance of the individual. The segmentation of the facial muscles -and in general of soft tissues- from CT scans is a harder issue and at the moment of the redaction of this manuscript no automated methods exist for addressing this difficult problem. In the context of project Facile, the work of L. Uro and Y. Maday will provide a statistical atlas model for the purpose of helping with manual muscle segmentation. For achieving this goal, they segmented the muscles manually.

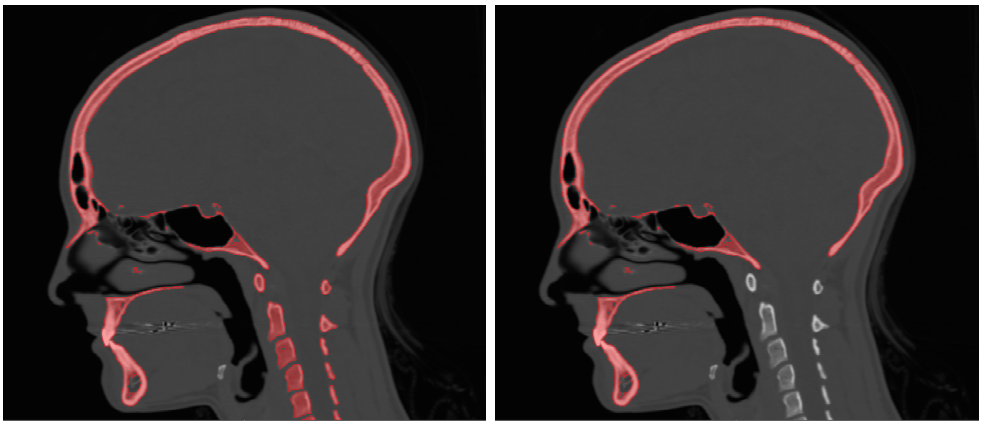


Figure 3: Segmentation of the bone (a) including column vertebral ; (b) after removing column vertebral.

6.4 $3d$ surface reconstruction using Marching Cubes

From the $2d$ binary images obtained from segmentation we aim at generating 3d mesh models of the skull, the face and facial muscles. The generation of a $3d$ triangular mesh from discrete volumetric data is in general referred as isosurface extraction [17]. The standard tool used for isosurface extraction is the *Marching Cubes* algorithm, introduced in [72]. This algorithm is in general employed for meshing an implicitly-defined domain. In its simplest version, the algorithm takes as input a cubic lattice $\mathcal{T}$ together with an implicit function ϕ defined at each vertex of $\mathcal{T}$. The function ϕ can take either positive value (if the vertex is up to the isosurface to extract) or negative otherwise. In our context, the lattice is given by the set of voxels in which the medical images are embedded and the binary function ϕ is positive at a given vertex if the vertex belongs to the label under consideration (bone, face or soft tissue), negative otherwise. The algorithm processes each cell independently in a divide and conquer paradigm. For each cell of $\mathcal{T}$ intersecting the isosurface $\phi = 0$, i.e. showing both positive and negative values of ϕ, the algorithm creates a piecewise linear approximation of the surface based on local informations. First, when an edge of $\mathcal{T}$ is composed of two extrema of opposite sign, a mesh vertex is created along the edge. In the simplest variant of the algorithm one can take the midpoint of the edge. Second, the previously defined mesh vertices are connected by triangles. Each element of $\mathcal{T}$ generates a surface patch composed of 1 to 4 triangles (Figure 4-(a)). The number of possible configurations (depending on the values of the label field on the current cell's corners) is 2^8, which reduces to 14 when quotienting for rotations and symmetries (see Figure 4-(a) for a few examples). The triangulation of the isosurface $\phi = 0$ is obtained by gluing the local patches. Despite the simplicity of the method, it is well known that in a few cases the classical Marching Cubes method can generate topologically ambiguous configurations or artificial holes, leading to an invalid triangulation. For example, an ambiguous situations occurs when a cell in $\mathcal{T}$ con-

tains a face in which vertex of different sign are opposite with respect to the diagonal two-by-two (see Figure 4-(b)(c)). In other words, the knowledge of ϕ at vertices of $\mathcal{T}$ is not enough to guarantee a fine reconstruction of the isosurface. For dealing with this ambiguities several variants of the algorithm have been developed [80, 75, 85]. Most methods are based on a further subdivision of $\mathcal{T}$, by using hierarchical data structures as octrees [117] or by tetrahedra subdivision [38], for estimating ϕ in the interior of the cell and solving the ambiguity. Also, the use of adaptative octrees allows to speed up the overall algorithm, generating an adaptively well sampled mesh [115]. This approach could avoid the need of decimating the resulting mesh in a post processing step. Finally we report the Extended Marching Cubes algorithm [66] for dealing with a finer approximation of sharp features as ridges or corners. For this work we used the Marching Cubes algorithm available in the Software *Amira* [98].

6.5 Geometric remeshing

As pointed out in the previous section, triangulations obtained from Marching Cubes procedure contain, in general, a prohibitive number of ill-shaped elements which are redundant and oversampled to correctly describe the geometry of the model. We are thereby confronted with the problem of *surface remeshing*. Surface remeshing relates to the enrichment -or oppositely the decimation- of a given triangulation $\mathcal{S}$ with the purpose of enhancing some quality requirements. What one means for 'quality' of a triangulation is intimately related to the application targeted and can involve vertex sampling, regularity, grading, complexity, elements size or shape. A remeshing procedure can either generate a new instance of the given triangulation or modify the existing one. Remeshing procedures are jeopardized in numerical simulations since very often the initial triangulation provided as support for a numerical computation suffers of very poor quality elements. Just to name a few situations in which remeshing is demanded:

- The initial triangulation is a very poor approximation of an underlying continuous surface and its geometry has to be enriched for the computation of differential quantities as normal vectors or discrete curvature;

- Surface models coming from scanning devices (as statues, architectural parts or of course medical images) generated with Marching Cubes like techniques are often oversampled and ill-shaped;

- Conversely, for real time computer graphics applications (e.g. character animation, rendering) it is essential to keep the number of element minimal, possibly by authorizing very stretched elements;

- The local element size has to be adapted to the numerical problem at hand, for example for capturing a moving interface in the context of a level set method;

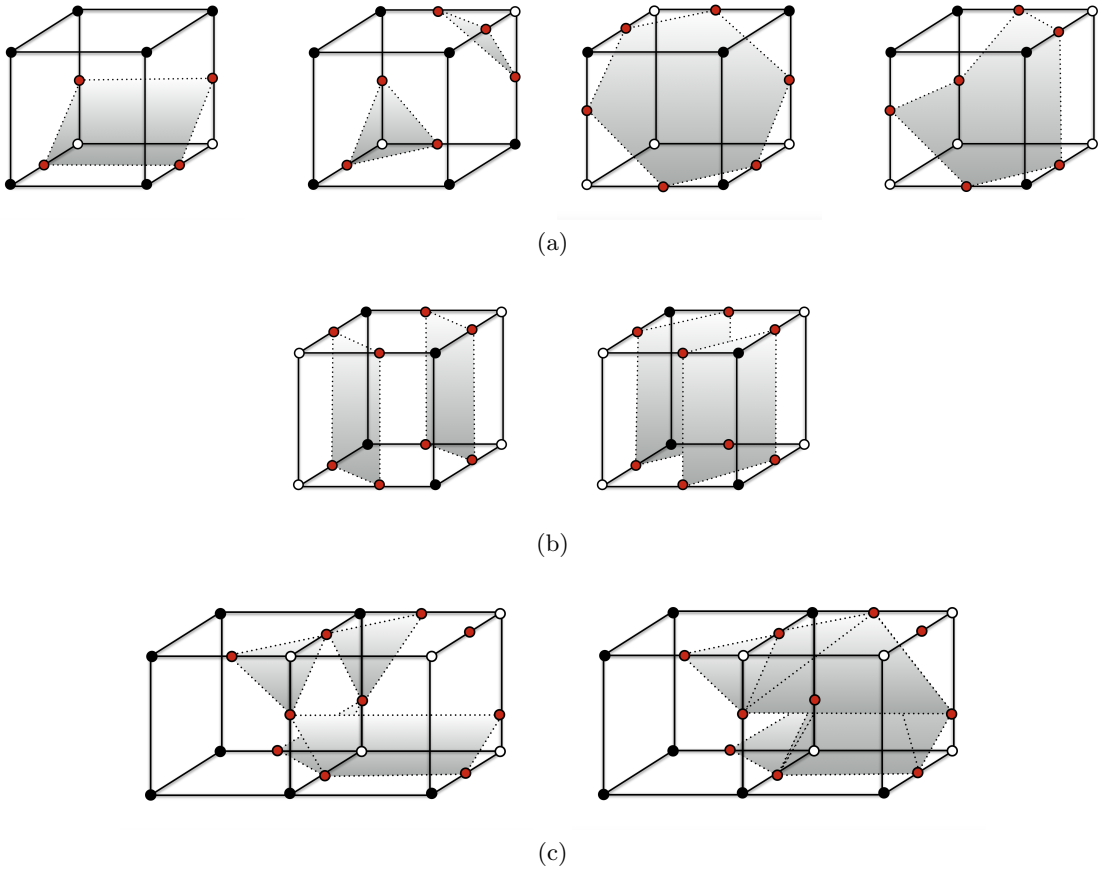


Figure 4: Some example of isosurface-cell intersection in the Marching Cubes algorithm. The white (resp. black) dots correspond to positive (resp. negative) values of ϕ. The grey zone corresponds to the reconstructed surface patch. The whole isosurface is constructed by gluing the local patches. (a) Some examples of configurations from the look-up table of Marching Cubes algorithm [72]. (b) A configuration from the look-up table of Marching Cubes algorithm [72] presenting a topological ambiguity. The grey zones correspond to two possible connections associated to the same configuration. (c) Topological incoherence generated from the look-up table of Marching Cubes original algorithm [72] (left) and repairing (right).

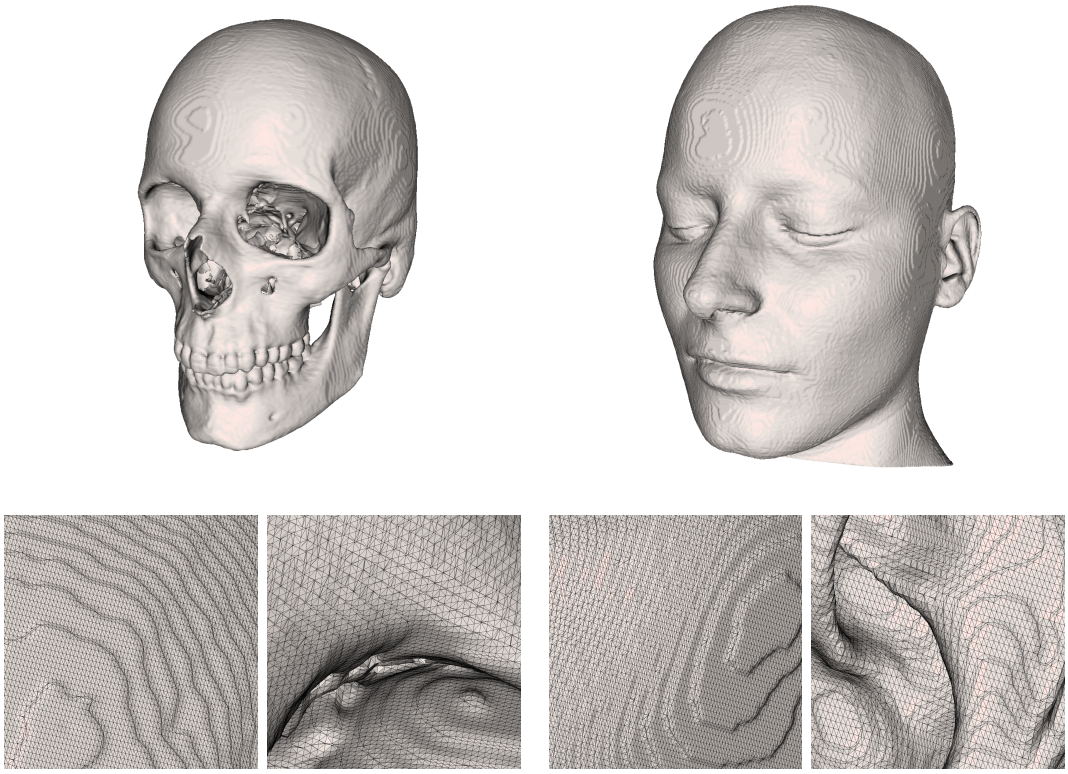


Figure 5: Top: 3*d* meshes of the skull (left) and the face (right) generated by Marching Cubes. Bottom: local enlargement showing the typical 'staircase' effect and almost degenerate triangles.

- Mesh elements have to fulfill a shape quality criteria since it impacts the numerical accuracy of the discrete solution computed using the mesh as geometric support [44, 14];
- The mesh is moved with the domain (Lagrangian deformation) and if no particular attention is paid to the deformation, the mesh degenerates in few iterations.

Generally speaking, the remeshing procedure aims at providing (i) a correct and accurate geometric approximation of the underlying 3d model (geometric mesh) and (ii) a computational mesh of hight quality elements suitable for finite elements simulations (computational mesh). Remeshing techniques are key ingredients of most engineering applications and geometric modeling tools; therefore there exists a huge literature devoted to this subject. Here we limit ourselves to discuss some very general ideas. See [5] for further references. Remeshing procedure can be categorized into *global* approaches, aiming at inferring from the initial triangulation a global description of the unknown underlying continuous surface, and *local* approaches, aiming at improving the mesh by applying local modifications. Among global approaches we can cite the

ones based on surface parametrization [3, 4, 92]. These methods aims at establishing a mapping between the initial supplied triangulation and a suitable domain [42]. In this work we relied on the second approach, using the remeshing procedure described in [30], and implemented in the surface remeshing software *mmg* [31]. Let $\mathcal{S}$ and $\tilde{\mathcal{S}}$ be respectively the initial and remeshed triangulations of the unknown underlying surface Γ. Together with the triangulation $\mathcal{S}$, the following parameters are taken into account:

- h_{hausd} controls the proximity authorized between $\tilde{\mathcal{S}}$ and Γ, so as to verify $d_H(\tilde{\mathcal{S}}, \mathcal{S}) < h_{hausd}$;
- h_{min} [resp. h_{max}] denotes the minimum [resp. maximum] allowed size for elements of $\mathcal{S}$;
- h_{grad} is a parameter that controls the mesh gradation, i.e. the size variation between neighboring elements.

The method relies on two key ideas. First, for ensuring proximity between the input and the new triangulations, the method guarantees that the Hausdorff distance $d_H(\tilde{\mathcal{S}}, \Gamma)$ stays smaller than a user or application prescribed given tolerance ε. For achieving this requirement the method computes *local* parameterizations in terms of Bézier surface patches, whose coefficients are determined from geometric quantities estimated from the source triangulation. This step aims at providing a fine reconstruction of the geometry of Γ and has a lot in common with [112]. Second, according to the inferred geometric information *and* the input user-defined parameters the method constructs an *isotropic metric size map*, i.e. a mapping $\Gamma \mapsto \mathbb{R}$, to which the final mesh $\tilde{\mathcal{S}}$ is adapted. See [44, 30] for more details about isotropic and anisotropic mesh adaptation to a metric size map. The improvement of the mesh quality is driven by performing local remeshing operators (Figure 6) on the initial triangulation. Figure 7 shows the influence of the h_{hausd} parameter over the remeshing procedure. For this study we kept the h_{hausd} parameter smaller than a prescribed tolerance of $0,1\%$ of the initial bounding box. This value is chosen according to the tolerance allowed in the segmentation process. As illustrated in Figure 7, the remeshing procedure also removes the 'staircases' artefacts due to the spatial discretization and connectivity. In this section we made the assumption that the triangulations supplied from image segmentation are *valid*. Note that in real-life applications, triangulations obtained by segmentation may often show intersecting faces, resulting in an invalid configuration. In Section 6.7 we describe a procedure based in implicit description of shapes whch allow for generating valid computational meshes from invalid data.

6.6 The role of landmarks in shape analysis for anthropology

As pointed out in Chapter 5 most of modern facial reconstruction digital methods account for the definition of sparse sets of 'meaningful' landmarks for describing and

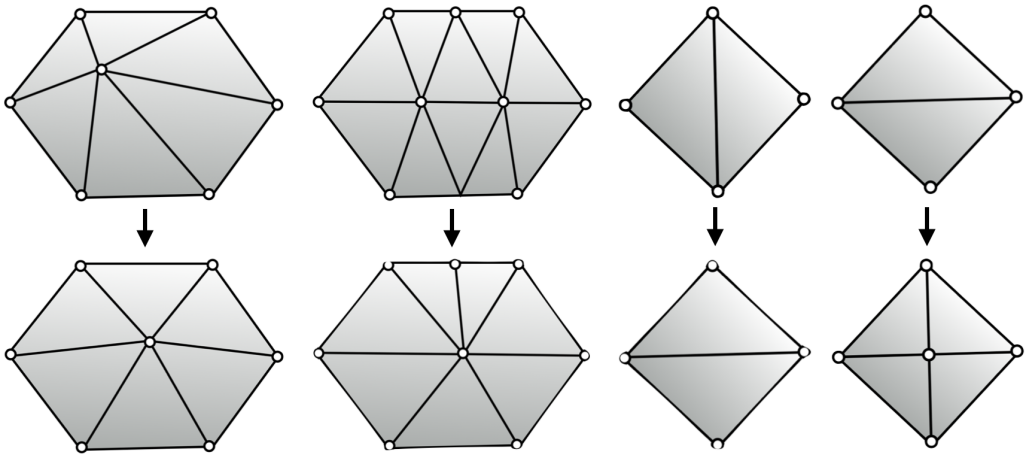


Figure 6: Local remeshing operators. From left to right: vertex relocation, edge collapse, edge flip, edge split.

comparing anatomical shapes. The quantitative analysis of shapes is referred in anthropology as Morphometrics. It is an active research field in physical anthropology and evolutive biology. Quantitative approaches to shape comparison have a long tradition. During the Renaissance period Leonardo da Vinci used to study proportions of the human body in terms of geometrical figures as circles, squares. Traditional morphometric approaches measure distances, angles, ratios. Geometric morphometrics was pioneered in the 80s in the seminal work [15]. The number of publications regarding landmark-based matching and comparison testifies the interest in this field [120, 53]. The main difference with respect to traditional morphometrics reposes on the representation of shapes in terms of coordinates of anatomical landmarks instead of linear measures. Landmarks play a fundamental role in computational anthropology and are defined as discrete anatomical loci that can be recognized as homologous in all specimens under investigation. In morphometric practices the sparse set of anatomical landmarks are used to build a statistical shape model and study variations. In facial reconstruction practices, landmarks are used to study intra-subject variations -by analysing the variations of the skull landmarks and their homologous face landmarks- and inter-subject variations [93]. Despite their crucial importance in applications as evolutionary processes, I consider that landmark-based shape analysis suffer of at least two major limitations. First, the choice of landmarks has to be done by an expert and should *a priori* account for the variation under consideration. In other words, a continuous and complex structure as a bone or an organ is reduced to a sparse set of points and if the choice of salient points is not well suited one may risk to loose information. Moreover, when the shape analysis is carried out on a big database, the manual setting of landmarks done by a trained executor becomes significantly time-consuming, with serious limitations to their effective application. For allaying both these issues some authors have developed landmark-free, fully automated methods for shape analysis. An example with application in paleontology

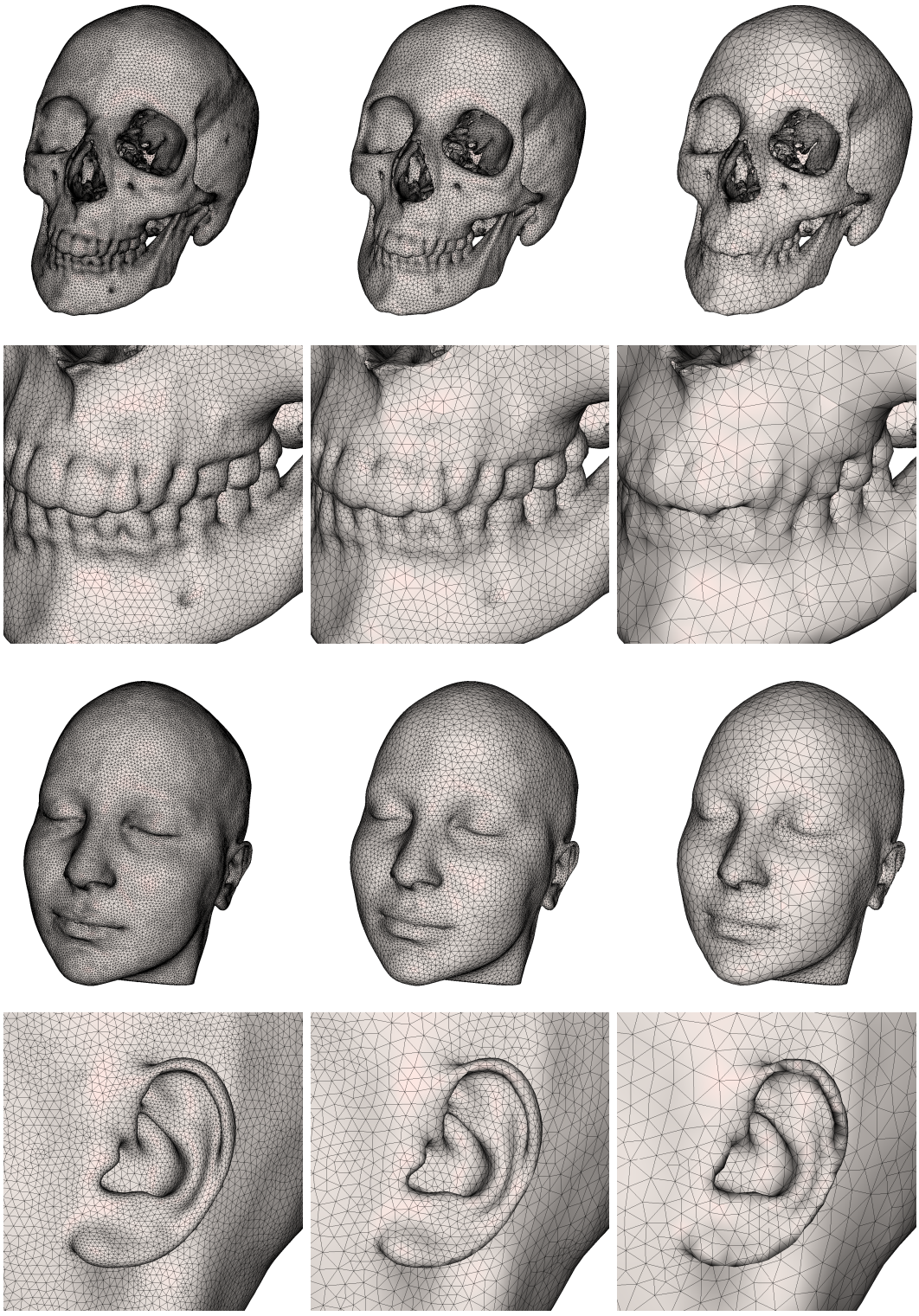


Figure 7: Curvature-dependent remeshing varying the Hausdorff parameter. Left: $h_{hausd} = 0.07$. Middle: $h_{hausd} = 0.3$. Right: $h_{hausd} = 1$.

is presented in [89]. Given the great advances in recent years in the fields of shape acquisition and analysis, and the possibility of collecting huge shape database, several very different works has been devoted to the definition of meaningful notions of shape similarity in the computer science and mathematic communities [76]. Next section presents a shape modeling procedure based on the shape matching method introduced in Chapter 2 as a promising framework for shape analysis and comparison.

6.7 Function-to-shape representation

Let us consider a shape database composed of closed surface triangulations $\mathcal{T}_1, \ldots, \mathcal{T}_N$ which are piecewise linear approximations of the boundaries of corresponding domains $\Omega_1, \ldots, \Omega_N \subset \mathbb{R}^3$. In the context of this study the shapes under consideration are the anatomical hard or soft tissue structures identified from medical images. The sole assumption we make on the initial discrete triangulations (and on the boundaries of corresponding shapes) is that they are closed and orientable, i.e. they define without ambiguity an interior (resp. exterior) region. Obviously for fulfilling the previous requirement, all the shapes must share the same topology. When the initial supplied triangulation is not closed -as in the case of the skull- one could apply the warping algorithm described in Chapter 4 for defining a close surface model. Distinct shape instances are not assumed to share the same number of elements and could possibly be invalid (showing for example intersecting faces). For the purpose of comparing and defining a notion of shape similarity we aim at defining a correspondence between shape instances from the sole knowledge of the mesh models. For doing this, for each shape Ω_i belonging to the dataset we build a function $u_i : \mathbb{R}^3 \mapsto \mathbb{R}^3$. Doing so we are able to parametrize the shape dataset in terms of N functions $u_1, \ldots, u_N$ which are more amenable for performing shape comparison, dimensionality reduction or for computing average shapes. Let us describe in details the procedure. Let us suppose that an initial rough alignment of the shape has been performed, so that all the considered shapes contains at their interior a 'small' subset ω which is clamped. Consider the common bounding box $\mathcal{D}$ (and corresponding computational mesh $\mathcal{T}_D$) in which all the considered shapes are embedded. Since we assumed the boundaries $\partial\Omega_1, \ldots, \partial\Omega_N$ to be closed, for each shape Ω_i we can compute at the vertices of $\mathcal{T}_D$ a piecewise linear approximation of the signed distance function d_{Ω_i} using the algorithm described in [33]. Note that this procedure can be performed even when the triangulation $\mathcal{T}_i$ is invalid. Let us suppose now that a reference shape Ω_0 together with a computational mesh $\mathcal{T}_0$ have been supplied. The shape Ω_0 has a fix topology and is clamped at ω. The reference shape can be either one instance of the database, or a geometric template as a sphere or an ellipse, or a template generated by combining the shapes in the database. We are thereby able to morph the reference template Ω_0 onto each instance shape, with global displacement u_i. Thanks to the matching process, we are able to describe all the shapes in the database in terms of the computational mesh $\mathcal{T}_0$ and the set of displacement fields $u_1, \ldots, u_N$. Each shape is defined

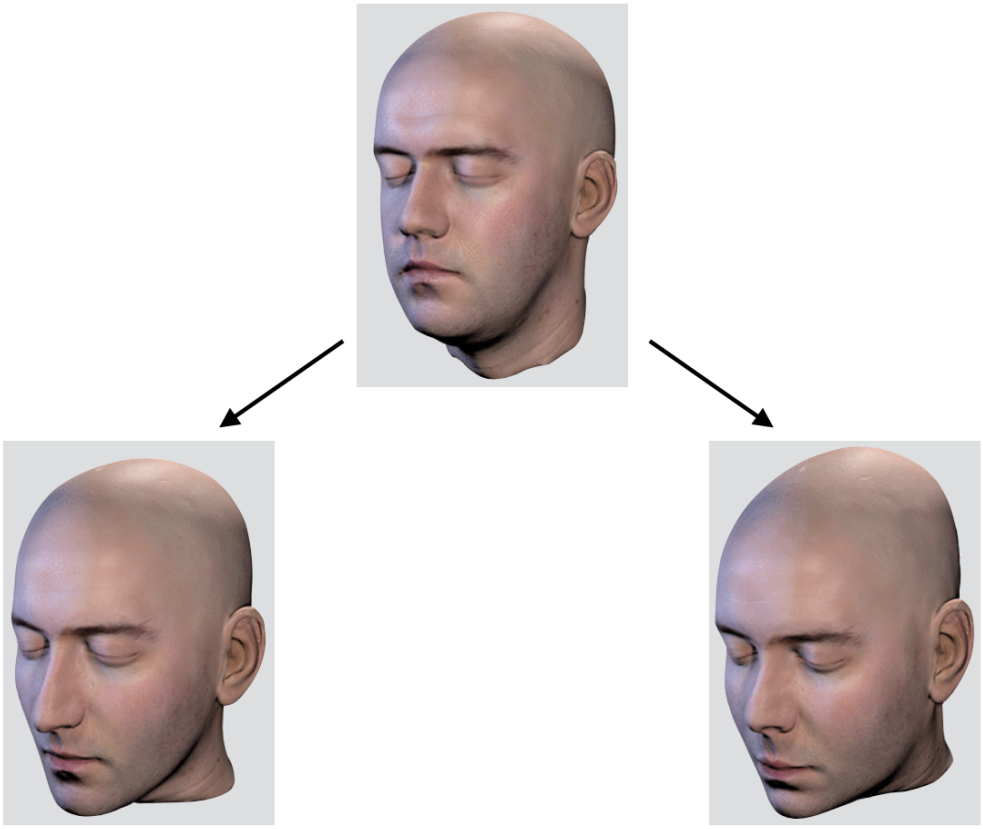


Figure 8: Texture transfer from a template template shape (top) to two instances of the database (bottom).

by the mapping:

$$\Omega_i = (I + u_i)(\Omega_0) \quad \forall i = 1, \ldots, N.$$

Equivalently, computational meshes $\mathcal{S}_i$ are obtained by pushing each vertex of the reference mesh according to :

$$\mathcal{T}_i = (I + u_i)(\mathcal{T}_0) \quad \forall i = 1, \ldots, N.$$

Note that now all the meshes share the same number of elements (vertices, triangles) and the same connectivity. See figure 9 for an example of such a procedure. This deformation may be used to transport quantities of interest from one shape to another, as in Figure 8.

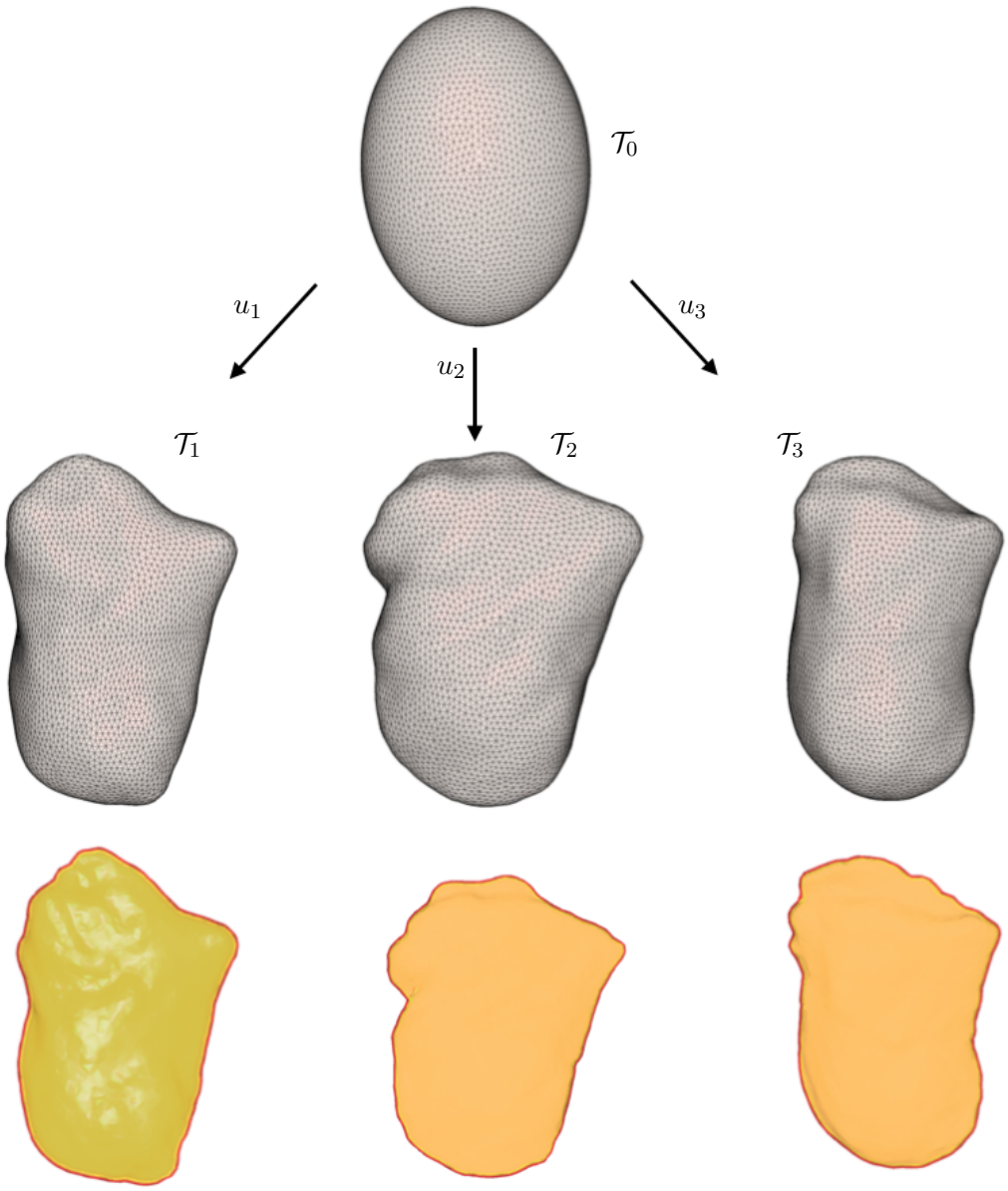


Figure 9: Reference mesh $\mathcal{T}_0$ (top) mapped onto three anatomical shapes in a database (bottom). Generation of the computational meshes $\mathcal{T}_1$,$\mathcal{T}_2$,$\mathcal{T}_3$ with the same number of elements and connectivity (middle).

Chapter 7

A landmark-free method for facial reconstruction

Contents

7.1 Introduction

This chapter contains the general workflow of the proposed method for Facial Reconstruction (FR). This is a joint work with Maya de Buhan.

Based on techniques of continuous deformation as morphing and warping, the proposed approach is a promising framework for integrating anthropological knowledge and for taking into account the biomechanical properties of soft tissue. The key features of our approach can be summarized in few points:

1. A database of coupled skull and face surfaces is generated from a CT scans data set;
2. An examination of the unknown skull determines anthropological parameters like age, sex, ethnicity and according to these parameters, some instances of coupled faces and skulls are selected among the craniofacial database;
3. For each selected individual, a $3d$ volumetric mask is generated for encoding the specific soft tissue variability. Each mask is composed of the region of space comprised between the inner bone surface and the outer face surface;

4. The inner surface of each mask (i.e. the bone surface) is mapped onto (a digital copy of) the unknown skull thanks to a landmark-free non-rigid dense surface morphing;

5. The considered mask is transported onto the unknown skull by deforming the facial template according to the previously computed skull mapping;

6. Starting from the collection of $3d$ masks, a set of estimates of the subject's face is generated. These facial shapes are combined together for proposing one -or multiple- instances of the reconstructed facial shape.

The main goal of a FR method is the estimation of the geometry of the facial shape from the skull. Eventually, the reconstructed face can be finalized by adding a texture for achieving more lifelike appearance. The proposed method has lot in common with previous deformation-based FR approaches [107, 91, 84], although the computational methods and the mathematical background used within each step of the process are very dissimilar. The chief peculiarity of the method -which was also the major concern motivating this work- is that it does not make use of anatomical landmarks. Limiting the need for manual procedures, we propose therefore a propitious framework for processing a huge amount of data in an automatic fashion. The sole first step of the process -which is standard in common FR methods, see Chapter 5- remains based on the intervention of an expert. Note that this step consists in an examination of the *sole* unknown skull; the manual effort required does not increase with the size of the database. The heart of our pipeline is the shape morphing method introduced in a much general context in Chapter 2. This procedure allows us to map each skull item onto the unknown skull. Due to its solid mathematical background, the method is able to guarantee that the reference and target shapes stay close from one other up to a fixed prescribed tolerance, which may be for instance inferred from the segmentation error. From the numerical point of view, this requirement is fulfilled by adjusting the size of the reference mesh elements (see Section 7.3). Also, the method allows us to establish dense correspondences between pairs of skull items, in the way described in Chapter 6. This correspondence combined with statistical methods can be used for clustering the individuals from the database into sub-population, according to morphological parameters. Finally, the method is used for computing new shapes from the combination of a set of existing ones (Section 7.5). This work is part of an ongoing multidisciplinary project. It must be viewed as the initial promising prototype that can be enriched and integrated with multidisciplinary expertise (from anthropology, medicine, biomechanics). See Chapter 8 for a discussion about difficulties, perspectives and openings of the method.

This chapter is organized as follows. Section 7.2 describes our choice for representing the unknown skull and the skull items of the database. Section 7.3 describes the process of skull matching. Examples and error computations are reported for showing the efficiency of the method. Section describes how the craniofacial template is adapted to the unknown skull. Eventually, Section 7.6 explains how to encode local information in the global deformation process.

7.2 Generation of a surface model of the skull

In the proposed procedure each skull template is described in three different ways. First, we generate a closed surface model which is 'as closer as possible' to the original skull model. This topology simplification issue is not novel in facial reconstruction methods (see [64, 105]) and is of crucial importance in the processus of matching individuals among the database (see section 7.3). The definition of the surface mesh model of the skull is achieved by iteratively deforming an elastic membrane using the algorithm described in Chapter 4. This procedure can be viewed as a $3d$ extension of the method proposed in [64, 105] for warping $2d$ CT scans images. The template shape well-tailored for our purposes is an annular ring enclosing the skull. Figure 1 depicts an example of skull warping. A comparison between the original skull triangulation and the deformed membrane warped onto the skull is depicted in Figure 2. The elastic parameters are the same used in Example 4.3.1. The process ends after $k_{max} = 53$ iterations (see Chapter 4 for a description of the parameters). The discrepancy between the original triangulation and the surface model is depicted in Figure 1-right, showing the capability of the method in recovering skull details as teeth, zygomatic and mental foramen. The recovering of the lower jaw is less accurate at level of insertion of the masseter muscle. For better recovering this zone one may consider to separate the lower jaw from the cranium during the segmentation process. This procedure includes also the separation of the teeth, which can be interesting for modeling facial animations. Thanks to the previous stage, we are able to compute the signed distance function to the boundary of the skull at the vertices of (the mesh of) a common computational bounding box. This implicit characterization will drive the process of elastic matching between skull templates. Level set functions for describing the skull face were firstly used in [109]. In our implementation we take advantage of an unstructured computational mesh for storing the values of the signed distance function. From the information of the signed distance function at its vertices, the initial computational mesh can be refined in the vicinity of the zero level set of the signed distance function and coarsened otherwise. See Figure 4 for an example of such a procedure. Finally, as argued in Chapter 6 we generate a set of valid computational meshes of the skulls in the database with the same number of elements and connectivity. Doing so, each skull is encoded by a displacement field u defined on the computational mesh of a reference skull triangulation (see Section 7.3).

7.3 Skull-to-skull registration

The deformation of the skull items onto the unknown skull is used for (i) establishing a correspondence between the shapes; (ii) deforming the database facial shapes onto the unknown skull. Therefore it is mandatory not to loose information during this process. For achieving this goal we rely on the use of dense meshes whose average element size is smaller than the amplitude between consecutive CT scans slices (0.7

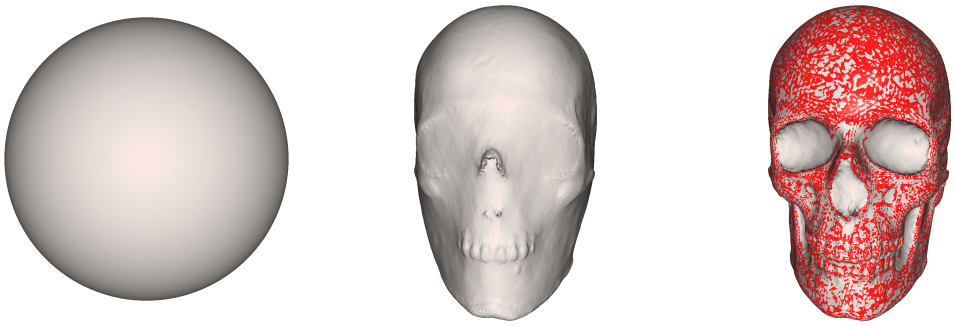


Figure 1: Left: template shape. Middle: advecting template shape. Right: discrepancy between the final shape and the target skull model.

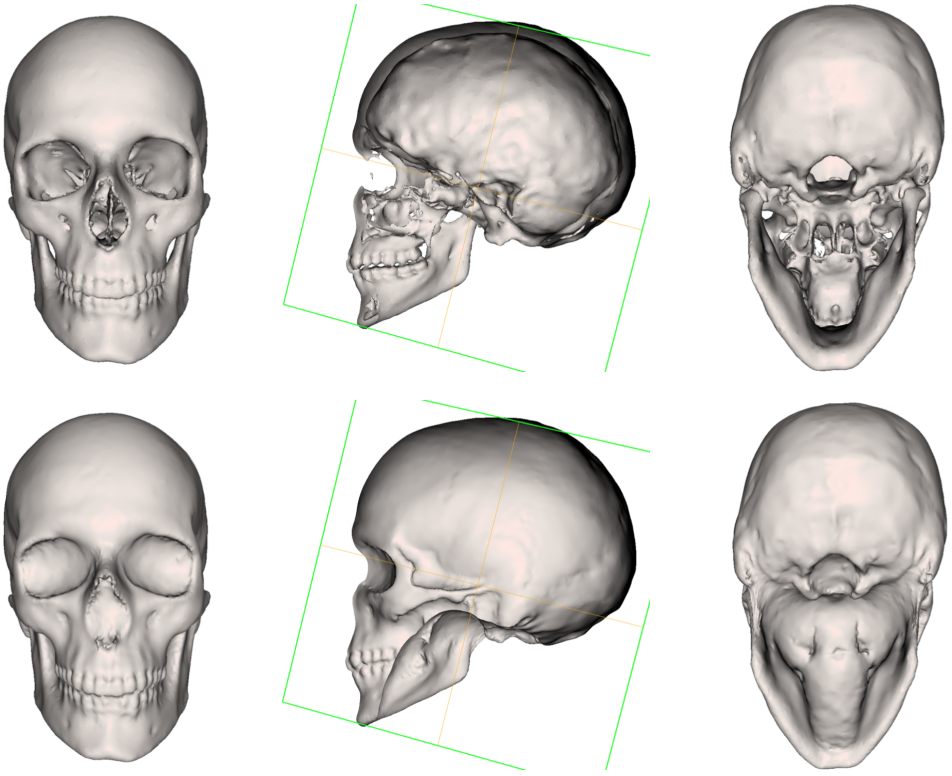


Figure 2: Top: 3d model of the skull. Bottom: wrapped skull.

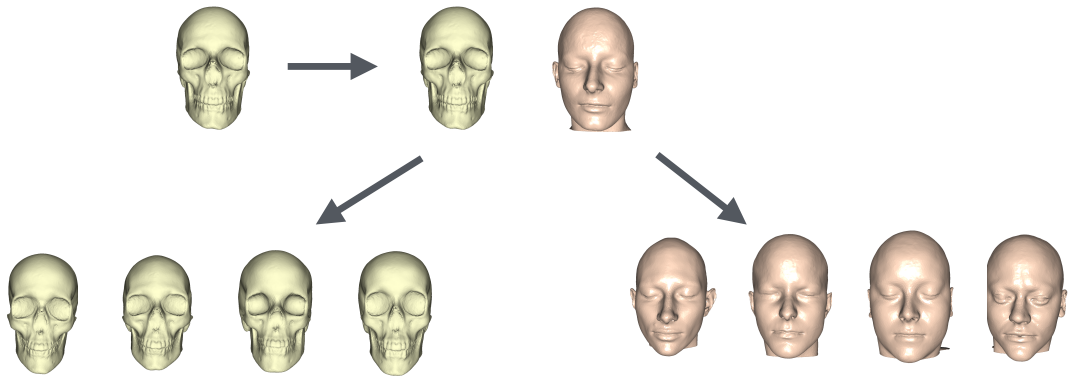


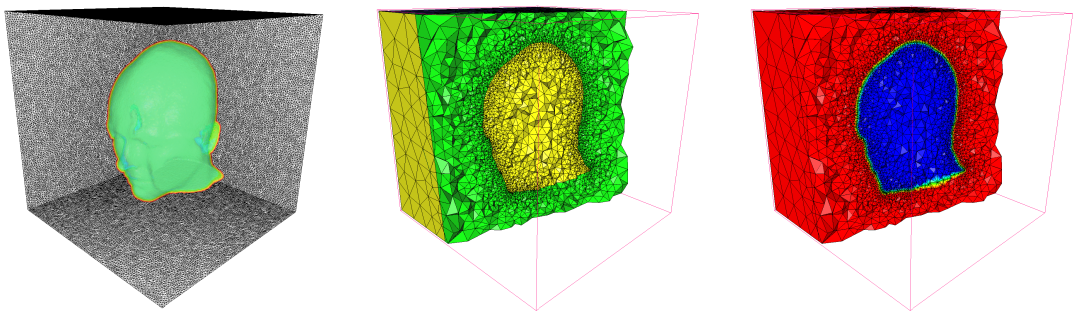
Figure 3: Matching skull and face templates within the database.

millimeters).

Let $(S_i)_{i=0,\dots,N}$ (and corresponding meshes $(\mathcal{S}_i)_{i=0,\dots,N}$) be the set of skull shapes from the database. At first, a specific reference skull, say $\mathcal{S}_0$, is randomly selected into the database. In a preprocessing stage, all the considered shapes are initially roughly aligned (without scaling) using an Iterative Closest Point (ICP) algorithm in order to guarantee the existence of a fixed common subset ω. After the rigid registration step, the reference skull $\mathcal{S}_0$ is matched onto all the $S_1, \dots, S_N$ skulls, producing a set of deformation fields $u_1, \dots, u_N$. Each field is defined over the region of space bounded by the skull envelope. The procedure of matching a reference skull onto the skull items is used to speed up the overall facial reconstruction procedure. Indeed, consider now the unknown skull S_T (and corresponding mesh $\mathcal{S}_T$). The link between S_T and the database skulls is inferred by the mapping:

$$S_T = \big(I + u_0\big)\big(I + u_i\big)^{-1}(S_i) \qquad \forall i = 1, \dots, N, \tag{7.3.1}$$

where u_0 is the global displacement mapping the reference skull S_0 onto the target skull S_T (see Figure 3). Therefore, since all the skull items of the database have been previously matched into each other, one only needs to compute the displacement u_0 to link S_T to each skull item. Figure 5 depicts two examples of skull matching. The template mesh $\mathcal{S}_0$ has about 130 000 triangles. The signed distance function to the target shape is computed at the vertices of a computational box which mesh is adapted to the contour of the target shape (see Figure 4-(b)). Note that in both examples the initial alignment between target and template shape is very rough and the two shapes are initially far from one others. For the test case of Figure 5-(a), the L^2 error $\mathcal{D}(\partial\Omega_{1300}, \partial\Omega_T)$ featured in equation (2.4.2) equals 0.2 millimeters, much smaller than the segmentation tolerance. For the test case of Figure 5-(b), the L^2 error $\mathcal{D}(\partial\Omega_{1250}, \partial\Omega_T)$ calculated on the boundary of the resulting shape Ω_{1250} equals



(a) (b) (c)

Figure 4: (a) Target shape Ω_T as the zero level set of the signed distance function; (b) Adaptive remeshing of the computational box; (c) Color map displaying the signed distance function to the target shape.

also 0.2 millimeters.

7.4 Face-to-face registration

The morphing chain described in the previous section for the skull is repeated for the facial shapes of the database. This step has not influence on the deformation of the facial templates, which can be performed independently. The face morphing is used in the final stage of the process for inferring to each facial shape the same number of elements and connectivity, and so for easily computing average shapes.

Let $(F_i)_{i=0,\dots,N}$ (and corresponding meshes $(\mathcal{F}_i)_{i=0,\dots,N}$) be the collection of facial shapes. The reference face F_0 (coupled with S_0) is deformed onto all the $F_1, \dots, F_N$ faces (with global displacements $v_1, \dots, v_N$).

See Figures 6 and 7 for examples of face matching. Again, both the target and the template meshes are embedded in a common computational box D. The template mesh $\mathcal{F}_0$ is the same for all the examples and has about 90 000 triangles. For the test case of Figure 6-(a), the L^2 error $\mathcal{D}(\partial\Omega_k, \partial\Omega_T)$ calculated on the boundary of the resulting shape Ω_{1200} equals 1.6 millimeters. For the test case of Figure 6-(b), the L^2 error $\mathcal{D}(\partial\Omega_k, \partial\Omega_T)$ calculated for $k = 1300$ equals 1.4 millimeters. For the test case of Figure 7-(a), the L^2 error $\mathcal{D}(\partial\Omega_k, \partial\Omega_T)$ calculated for $k = 1200$ equals 1.6 millimeters. For the test case of Figure 7-(b), the L^2 error $\mathcal{D}(\partial\Omega_k, \partial\Omega_T)$ calculated for $k = 1150$ equals 1.8 millimeters.

Remark 7.4.1. *Due to their flexibility, in some case the ears deformed too drastically, leading to subdomain intersections which prevent the convergence of the overall matching process. In order to avoid troubles, and since the ears are not linked with the underlying skull morphology, we decided to not include the ears in the face matching*

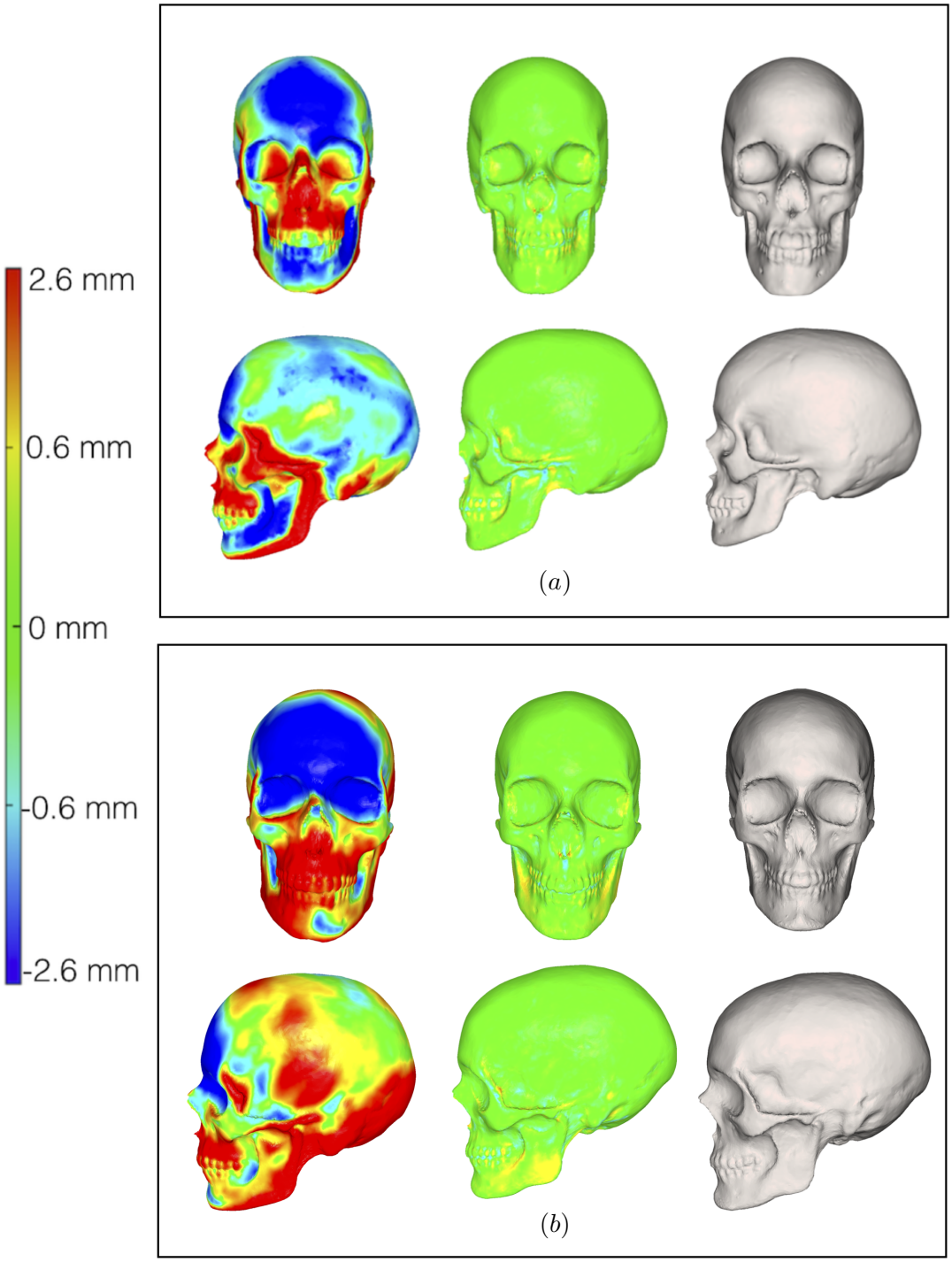


Figure 5: Left: Template skull shape S_0 and discrepancy w.r.t. the target shape. Middle: deformed shape S_0^k at the end of the procedure and discrepancy w.r.t. the target shape. Right: target shape.

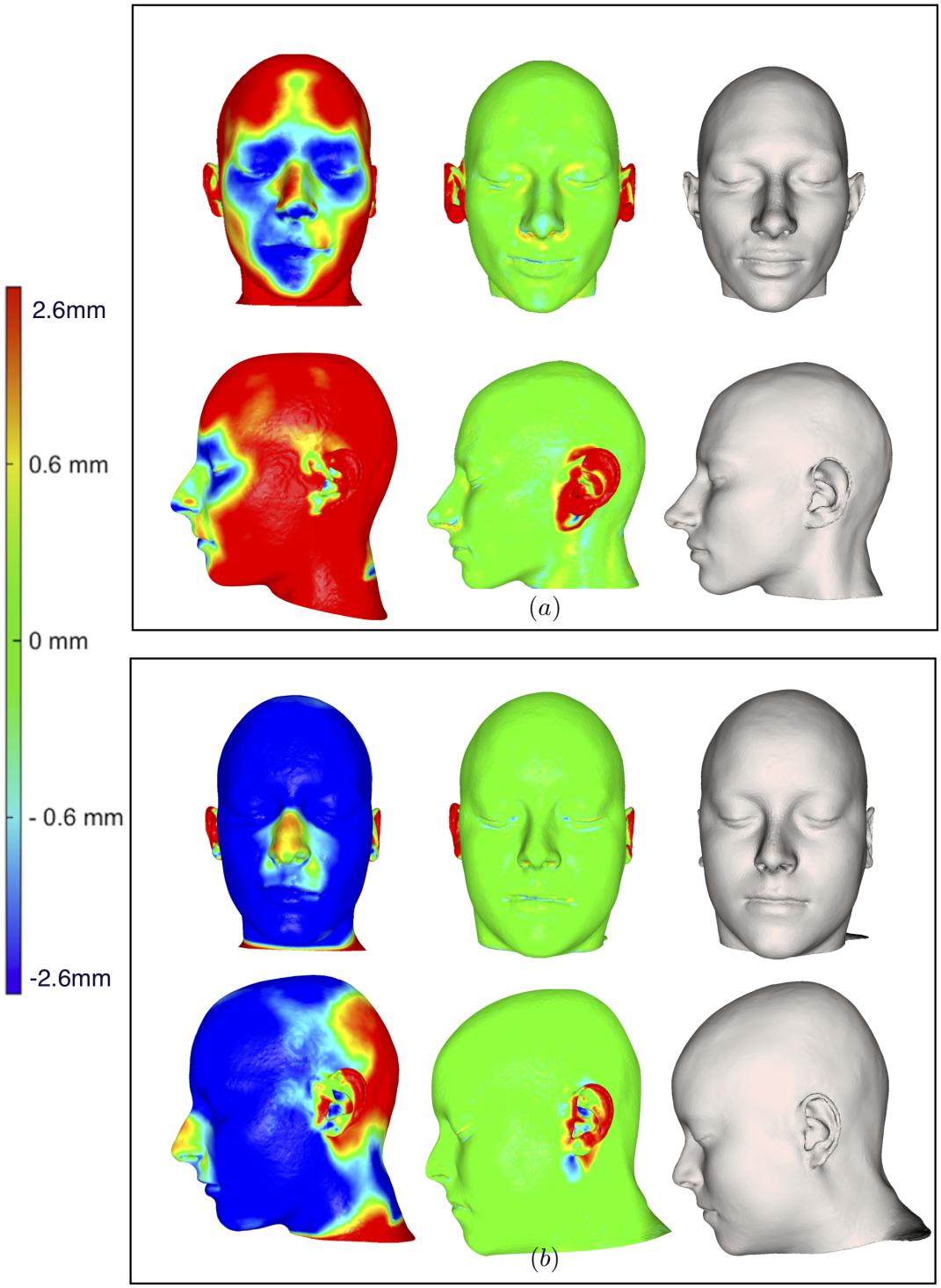


Figure 6: Left: Template facial shape F_0 and discrepancy w.r.t. the target shape. Middle: deformed shape F_0^k at the end of the procedure and discrepancy w.r.t. the target shape. Right: target shape.

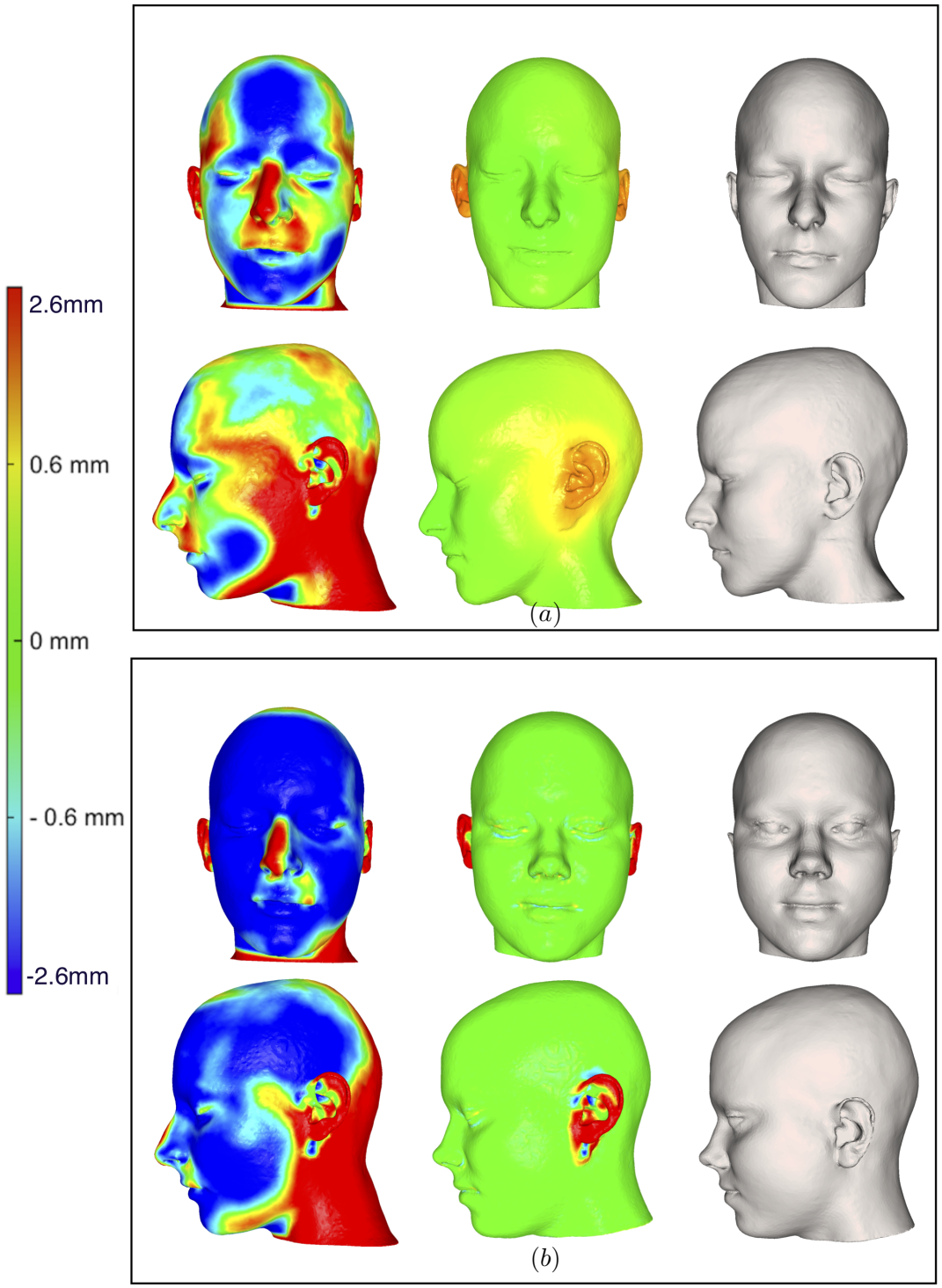


Figure 7: Left: Template facial shape F_0 and discrepancy w.r.t. the target shape. Middle: deformed shape F_0^k at the end of the procedure and discrepancy w.r.t. the target shape. Right: target shape.

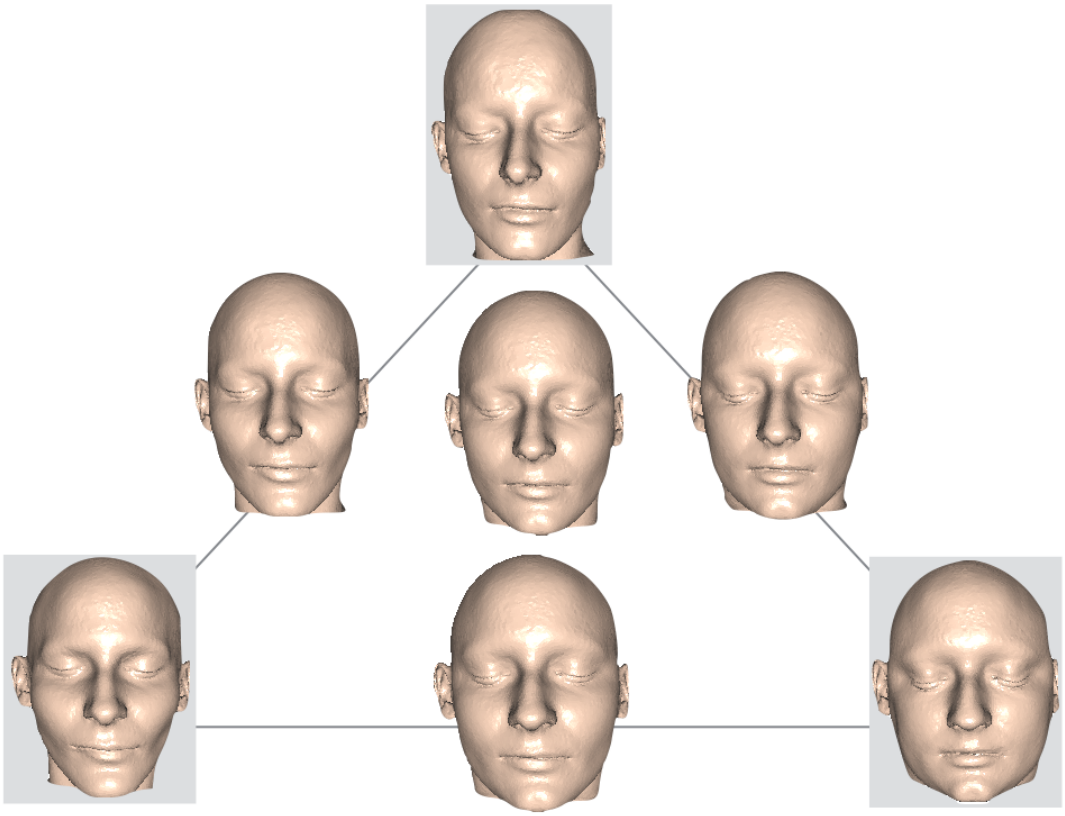


Figure 8: The corners of the triangle display three face templates. The top template is chosen as reference shape and matched into the bottom right template and the bottom left template. The middle of the triangle and the mid-edges display the 'barycenter' face computed by averaging the displacement fields.

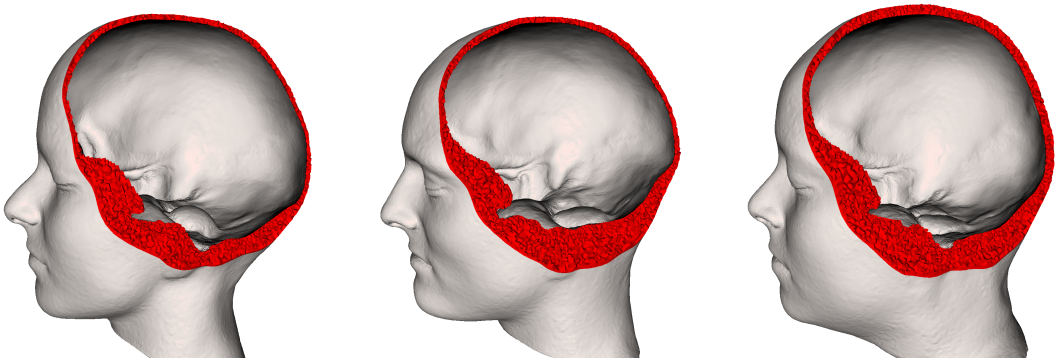


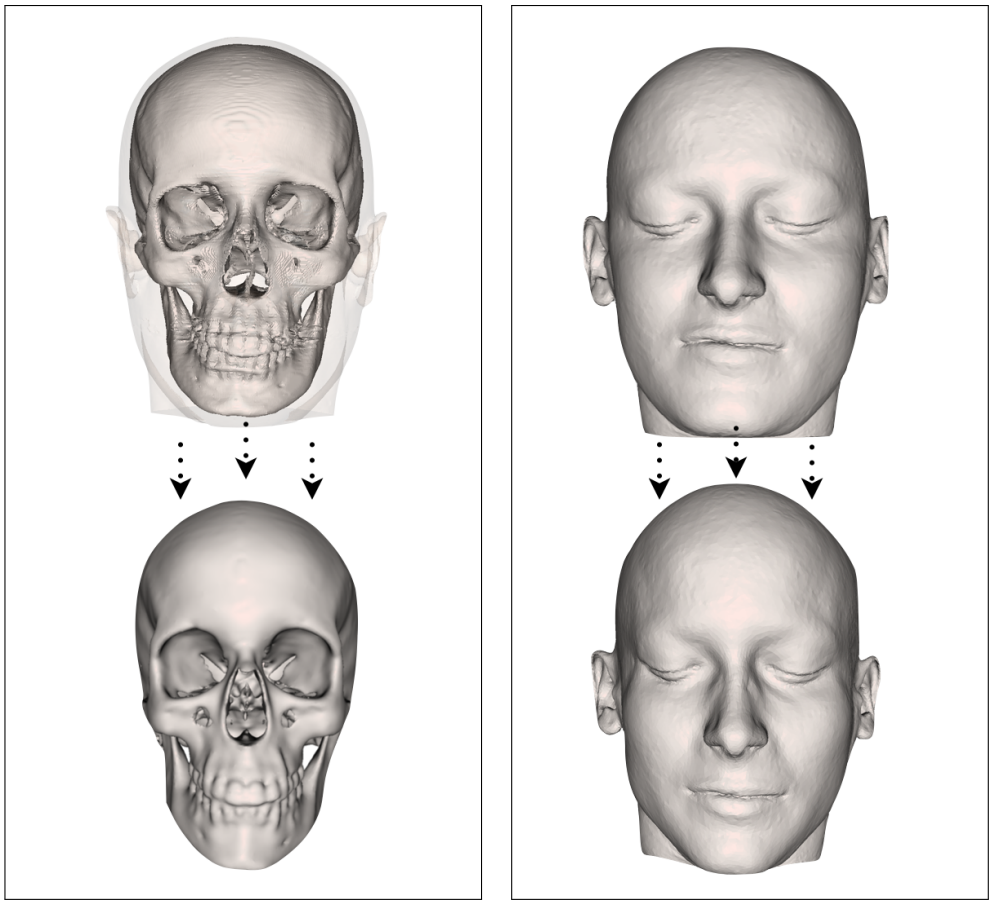
Figure 9: Three craniofacial templates.

process. This is accomplished by defining a surface patch $\Sigma \subset \partial\mathcal{F}_0$ *which is excluded by the minimization process. Due to this choice, note that in correspondence of the ears the accuracy of the matching is no longer guaranteed. This discrepancy explains the bigger error we get in face matching with respect to skull matching. When computing the discrepancy* $\mathcal{D}(\cdot, \partial\Omega_T)$ *on the sole region* $\partial\Omega_k \setminus \Sigma$ *of the advected domain* Ω_k *we get an error which order is close to the one registered for the skull matching (between* 0.2 *and* 0.3 *millimeters).*

Also, thanks to the matching process, we can generate new facial shapes by averaging a set of existing shapes. See Figure 8 for an example of such a procedure.

7.5 Model to skull registration

This step of the procedure refers to the deformation of the craniofacial template onto the unknown skull for reconstructing an estimate of the unknown face F_T. The procedure is accomplished in three steps. First, the volume between the coupled surface meshes $\{(\mathcal{S}_i, \mathcal{F}_i)\}_{i=0,\dots,N}$ is meshed with tetraehdra and filled with an elastic material, generating 3d masks $(\mathcal{M}_i)_{i=0,\dots,N}$ which incorporate the soft tissue information (see Figure 9). For the generation of the tetrahedral mesh we used the open source software *Tetgen*. Then, as explained in Section 7.3 the global displacement u_0 mapping the reference skull S_0 onto the target skull S_T is computed and used for linking S_T to the database. Then, thanks to the mapping of the skull item S_i onto the unknown skull S_T, the 3d mask is elastically deformed under the effect of the boundary changes (see Figure 10). This step is accomplished by computing the solution of an elastic problem defined on $\mathcal{M}_i$. Eventually, all the 'transported' faces are averaged according to the skull similarities.

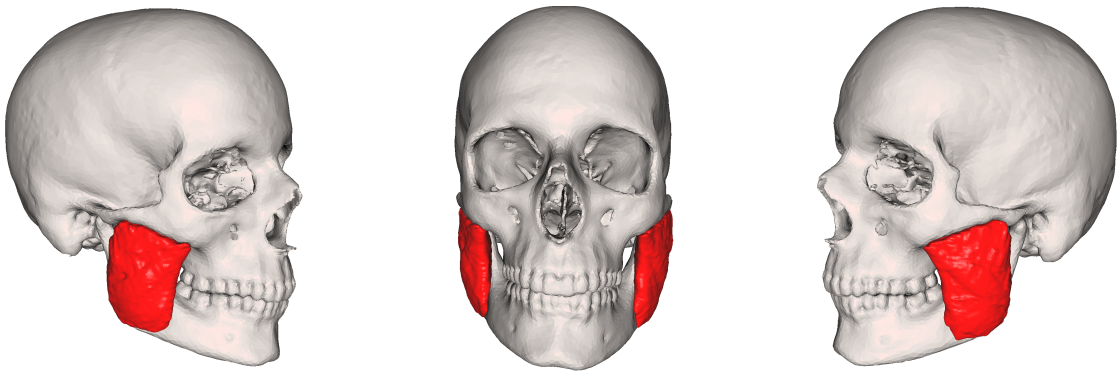


(a) (b)

Figure 10: (a) Deformation of a skull item onto the unknown skull. (b) Deformed craniofacial template under the effects of skull changes.

7.6 Enclosing local information

The approach proposed in this section is part of a joint work with Lydie Uro. The proposed method for facial reconstruction deforms the soft tissue mask in bulk onto the unknown skull. This global deformation can be integrated by adding the local information and improving the accuracy of the prediction. For instance, imagine that instead of the sole skull one has informations about some facial muscle. For explaining the idea, let us focus on the masseter muscle, which fill an important volume in the soft tissue and it is relevant from a functional point of view. Suppose also that the same information (shape and location of the masseter muscle) is available for all the individuals of the database. Then, one can perform exactly the routine described in Section 7.1 using the couple skull/muscle instead of the sole skull. Since the masseter muscle rests on the skull support (see Figure 11-(a)), when discretizing with meshes



(a)

(b)

Figure 11: (a) $3d$ model of the skull equipped with the masseter muscles (colored in red). (b) Warped model.

we obtain two surfaces (respectively, the warped skull and the muscle which is a priori closed) intersecting in the region of contact between the muscle and the skull. For avoiding the need of modifying and repairing this invalid model, we rely on the warping algorithm described in Chapter 4. The considered source triangulation in this case is the couple skull and muscle. The result of such a procedure is depicted in Figure 11-(b). In Chapter 8 we will show with an example the influence of the knowledge of the muscle masseter over the whole FC process.

Chapter 8

Results and perspectives

Contents

8.1 Introduction

In this conclusive chapter I present a gallery of facial reconstruction results obtained with the method described in Chapter 7. The rendering of the reconstructions has been realized by Loic Norgeot.

The proposed results have to be handled carefully. At time time of writing we are still developing our strategy for FR and in particular we did not devise a *validation* protocol. Therefore, we are not able to 'rigorously' evaluate the quality of the reconstructions. However, we content ourselves with a 'visual' inspection of the results. The chapter ends with a discussion about perspectives.

8.2 Test cases

For a first prototype of our method we started with a 'simple' assignment: given an individual from the database -for which both skull and face are known- use the proposed FR method to estimate the face starting from the observation of the database and the skull. Doing so, we are able to compare the estimated face with the original one. Let us consider the example depicted in Figure 1. A collection $(F_i)_{i=0,\dots,N}$, $(S_i)_{i=0,\dots,N}$, with $N = 8$, of faces and corresponding skulls is selected from the database. Owning to the strategy proposed in Chapter 7, the set of faces is deformed onto the 'unknown' skull S_T (again, selected from the database) producing

a set of N estimates $(F'_i)_{i=0,\dots,N}$ of the unknown face. For the sake of simplicity, we propose in Figure 1-(a) the geometric average of the set of facial estimates of the original face (reported in 1-(b)). The average is computed using the procedure of Section 6.7. The discrepancy $\mathcal{D}$ between a facial shape Γ and a reference facial shape Γ_0 is evaluated by the following mean error:

$$\mathcal{D}(\Gamma, \Gamma_0) = \left(\frac{1}{|\Gamma|} \int_{\Gamma} d^2(x, \Gamma_0) dx \right)^{\frac{1}{2}},$$

where $|\Gamma|$ is the measure of Γ and $d(\cdot, \Gamma_0)$ is the Euclidean distance to Γ_0. This error estimator is used for evaluating the vicinity between:

1. the skull templates of the database and the unknown skull;
2. the face templates of the database and the unknown face associated with the unknown skull;
3. the face templates after deformation onto the unknown skull and the unknown face associated with the unknown skull;
4. the face templates of the database and the reconstructed average face.

The minimal, maximum and mean values of $\mathcal{D}$ for the 8 templates of the database are reported in Table 8.1. In particular, we observe that the discrepancy between the face template and the unknown face is smaller after deformation for all the individuals. Thus, the elastic transformation used in the method seems to be a good tool to transport the faces close to the unknown face. Moreover, the face showing after deformation the smallest discrepancy w.r.t. the unknown face (2.93 mm) corresponds to the skull showing the smallest discrepancy w.r.t. the unknown skull (5.44 mm). Finally, the discrepancy between the unknown face and the average predicted face is 4.24 mm. It is smaller than the discrepancy for any individual in the database, so this measure $\mathcal{D}$ can be used for an automatic numerical identification. Figure 2 shows the distribution of the error over the reconstructed surface. A positive value of the error means that the estimate is up the original face; otherwise, a negative value corresponds to regions in which the original face is underestimate. Note that in the regions of interest (excluding ears and neck in particular), the error is less than a millimeter. We also remark that in these regions we overestimate the thickness of the tissue. Using the information coming from the BMI of the individuals in the database could be a way to propose different reconstructed faces corresponding to different BMI of the unknown individual.

Next, the same test case is studied using the approach featured in Section 7.6. We consider the same set of faces and skulls which are now equipped with their masseter muscles. The result of this procedure is depicted in Figure 1. The discrepancy between the estimate and the original face is displayed in Figure 2-bottom. Note the improvement of the reconstruction in the regions of the face face close to the masseter muscle.

(a) (b) (c)

Figure 1: (a) Reconstructed face obtained by averaging 8 templates from the database (without enclosing local information). (b) Reconstructed face obtained by averaging 8 templates from the database (enclosing local information). (c) Original face.

Eventually, let us consider the whole collection $(F_i)_{i=0,\dots,N}$, $(S_i)_{i=0,\dots,N}$, with $N = 26$, of shapes in the database. Figures 3,4, 5, 6, 7 depict the reconstructions of five faces which are successively removed from the database. Again, the proposed reconstruction is obtained as a geometric average of the $N = 25$ estimated faces. Figures 8, 9 show the distribution error for the five test cases. Note that the reconstruction featured in 6, even showing a visual resemblance with the original face, fails to correctly estimate the correct soft tissue. This example suggest that it would be preferable to propose multiple reconstructions (corresponding to several BMI) instead of a mere single one. Also, note the underestimation of the nasal region when the original face shows a prominent nose as the one featured in Figure 7. This result is due to the fact that the considered nasal shape is not represented in our database. Using information coming from a larger database could help in improving the result.

Discrepancy between $\cdots$	min	mean	max
1. skulls of the database and the unknown skull	5.44 mm	7.86 mm	13.46 mm
2. faces of the database and the unknown face	5.02 mm	8.94 mm	15.69 mm
3. faces after deformation and the unknown face	2.93 mm	5.8 mm	10.35 mm
4. faces of the database and the average predicted face	4.54 mm	7.03 mm	11.28 mm

Table 8.1: Discrepancy between shapes for our test case.

8.3 Perspectives

The proposed reconstruction method reposes on the 'physical' deformation of templates of coupled faces and skulls onto the unknown target skull. Even if our

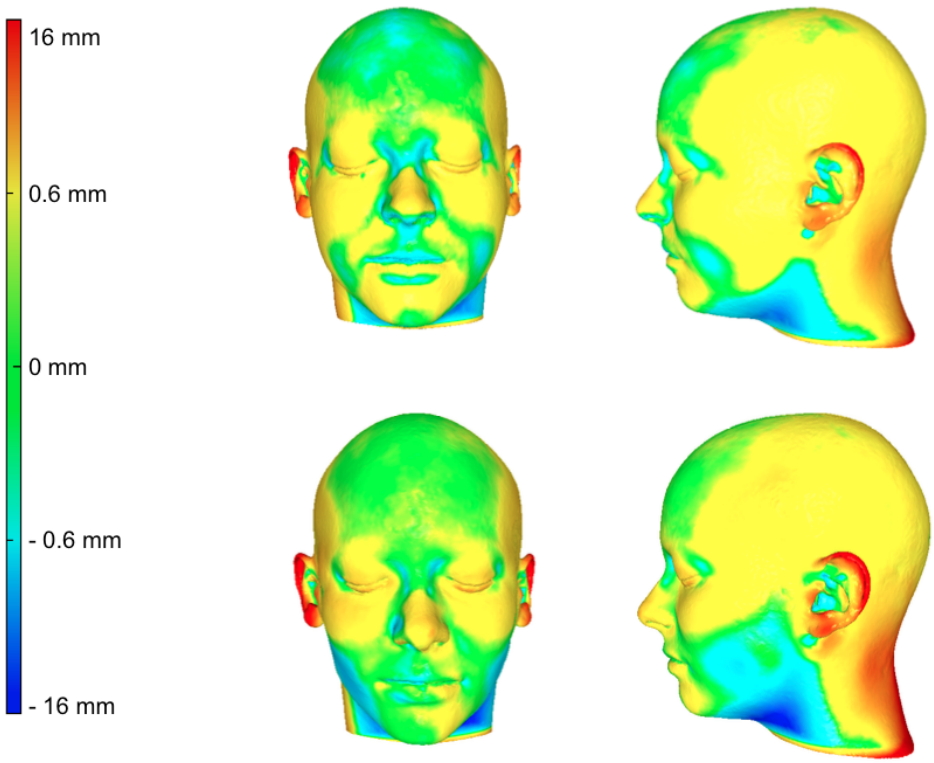
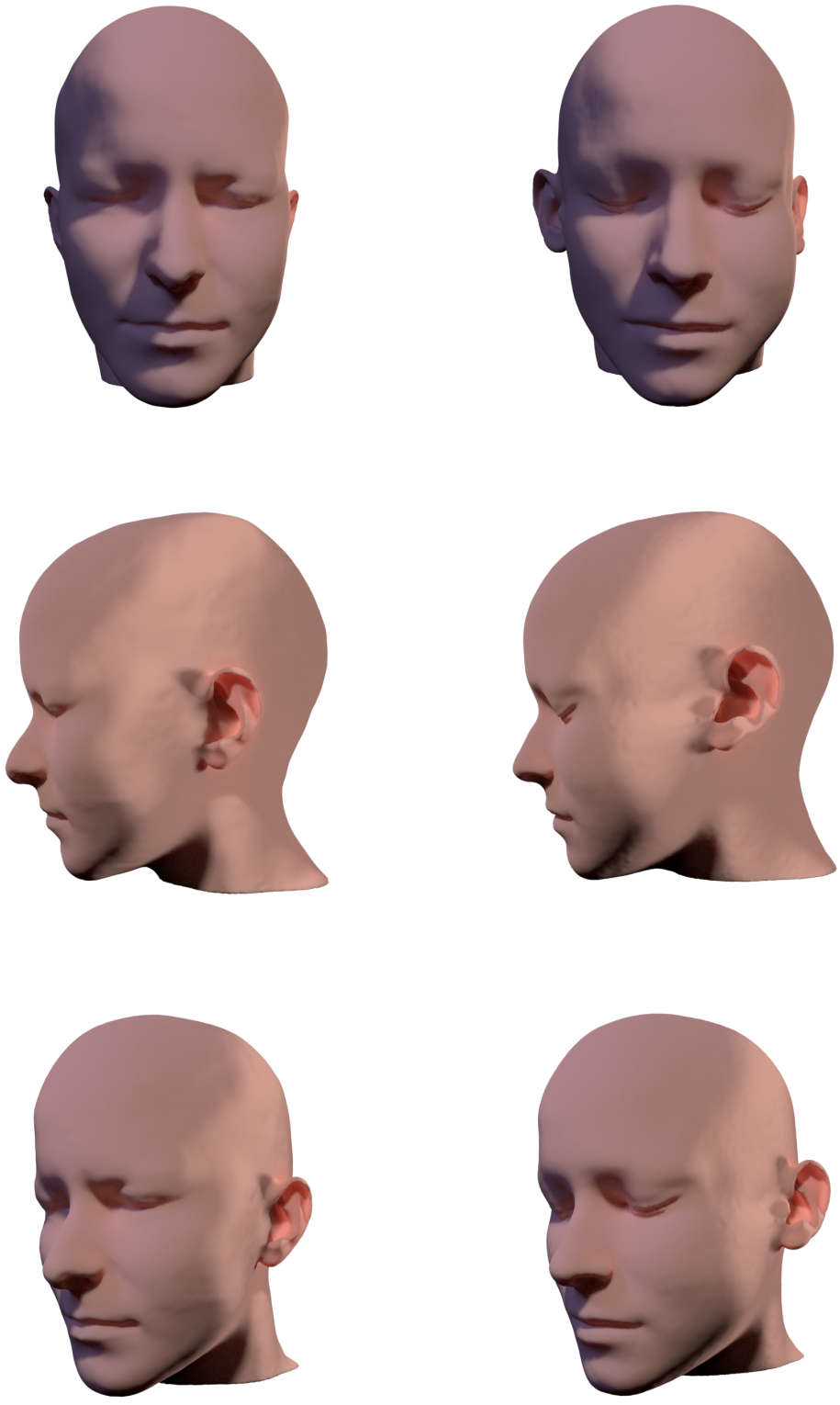
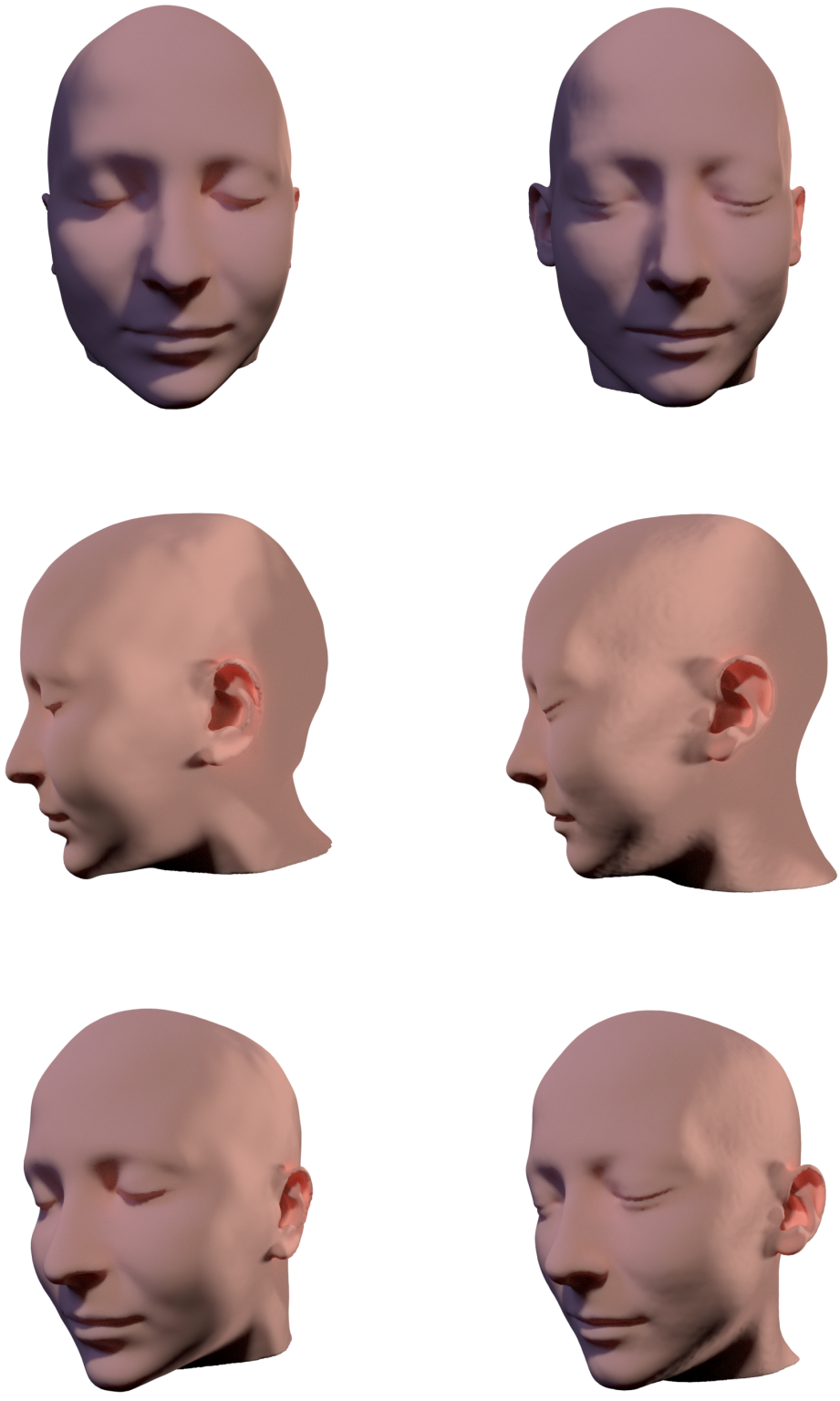


Figure 2: Discrepancy between the predicted and the original faces for the test case of Figure 1 with (bottom) and without (top) enclosing local information.



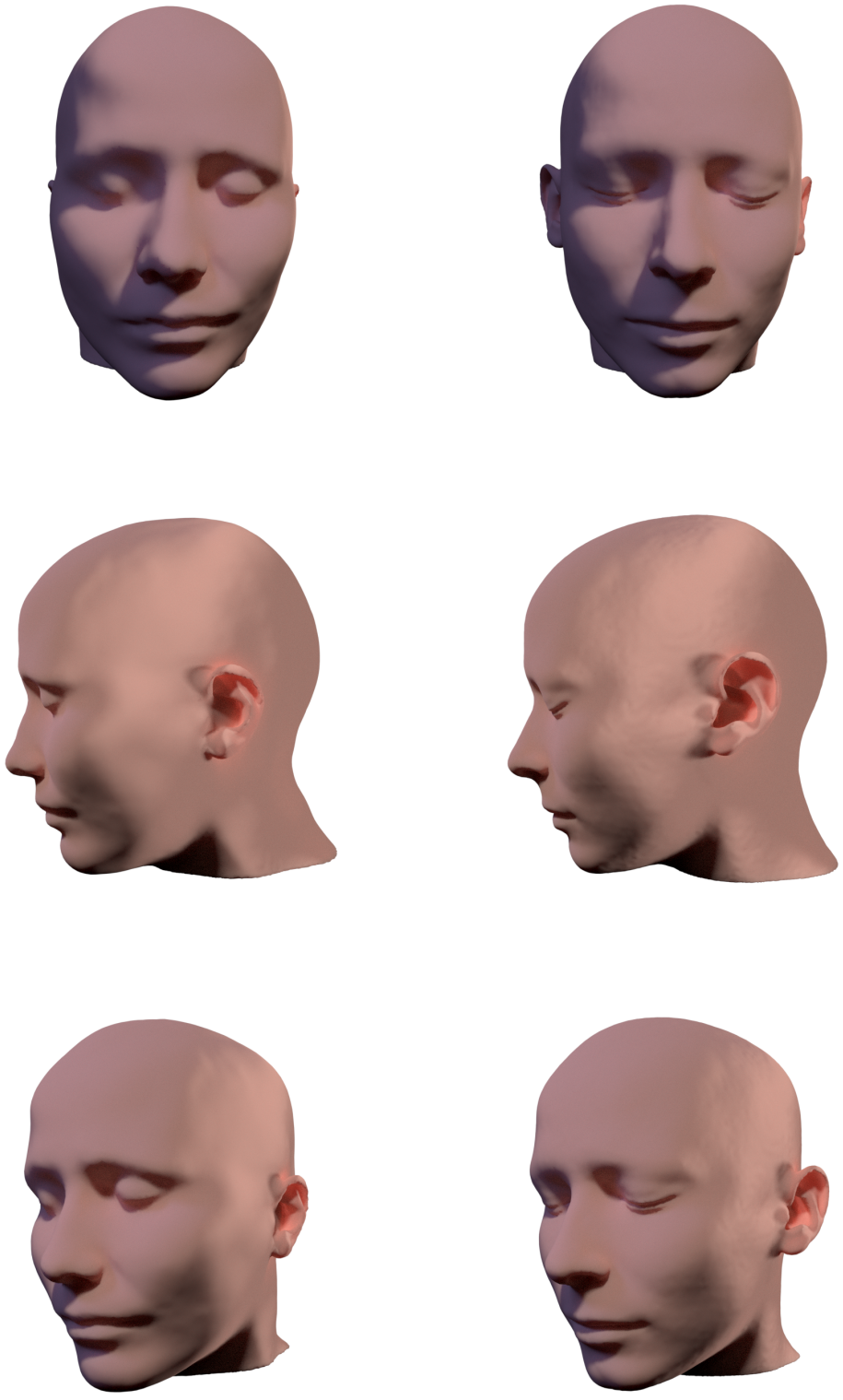
(a) (b)

Figure 3: (a) Original face. (b) Reconstructed face.



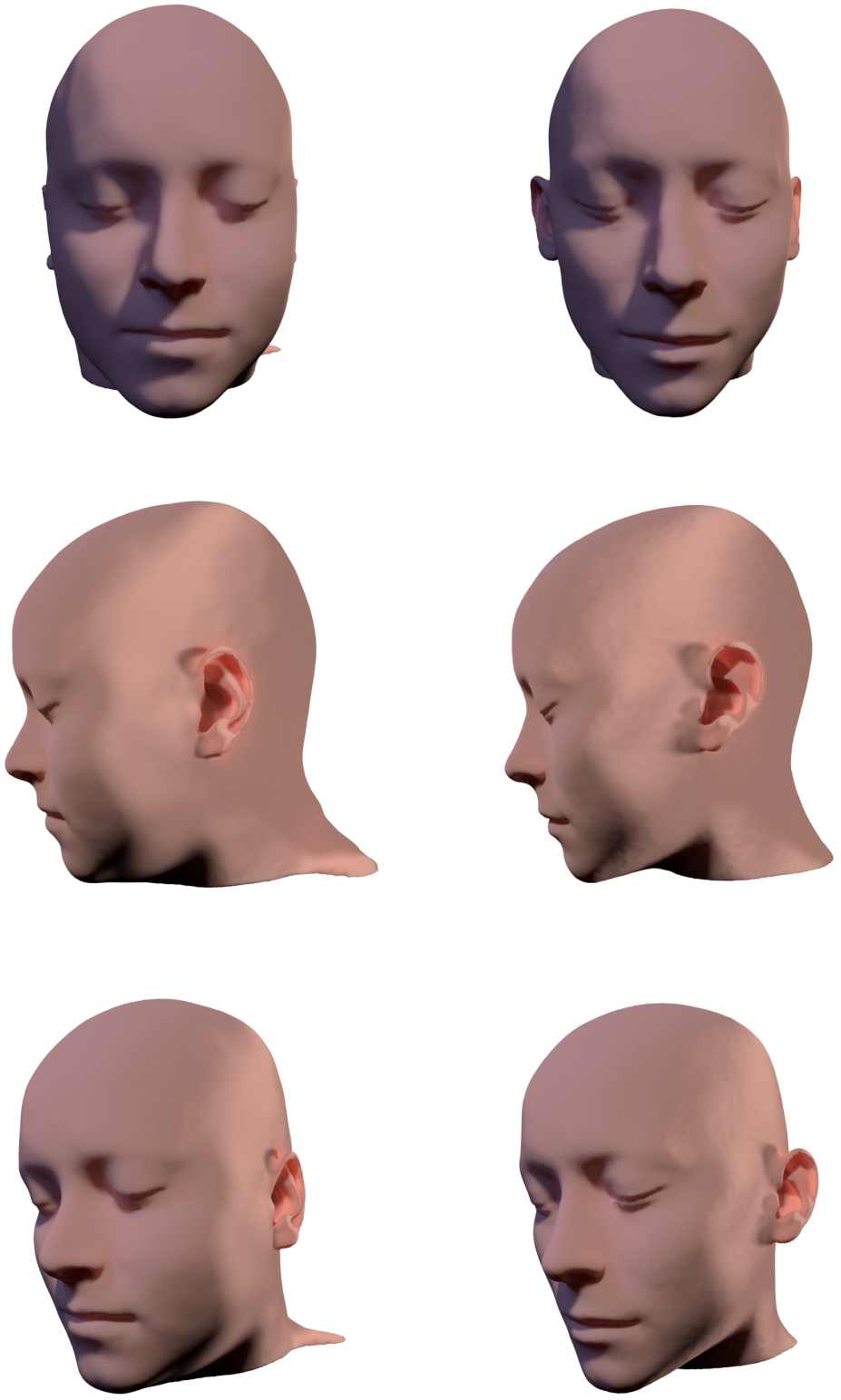
(a) (b)

Figure 4: (a) Original face. (b) Reconstructed face.



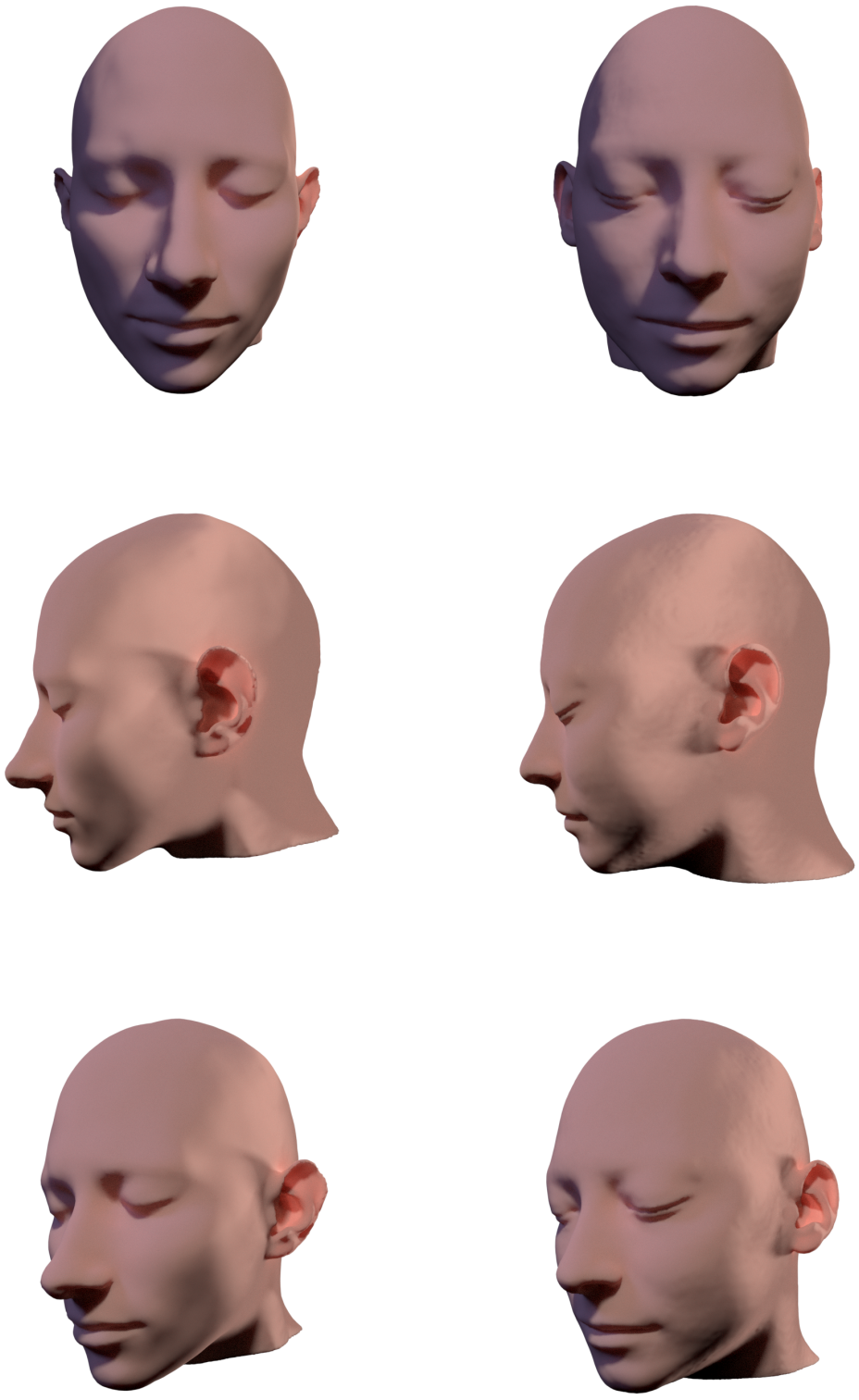
(a) (b)

Figure 5: (a) Original face. (b) Reconstructed face.



(a) (b)

Figure 6: (a) Original face. (b) Reconstructed face.



(a) (b)

Figure 7: (a) Original face. (b) Reconstructed face.

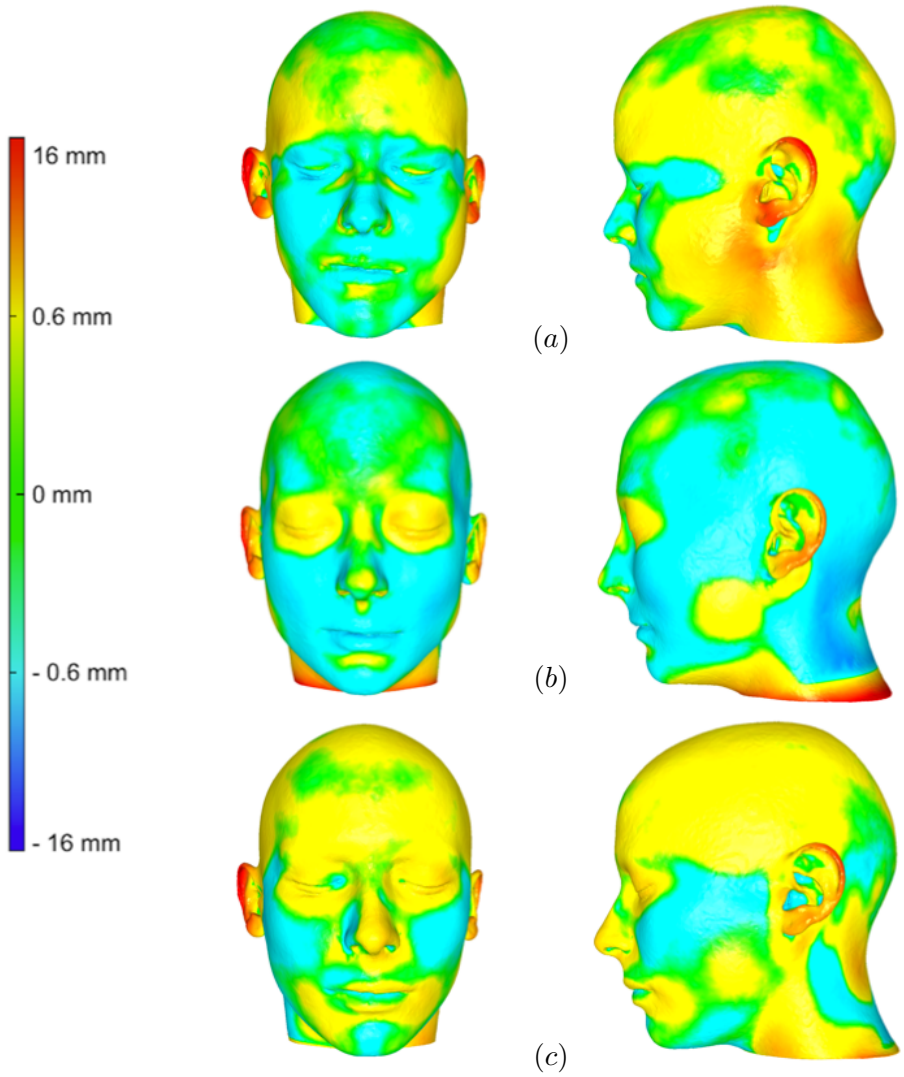


Figure 8: Discrepancy between the predicted and the original faces (a) for the test case of Figure 3; (b) for the test case of Figure 4; (c) for the test case of Figure 5.

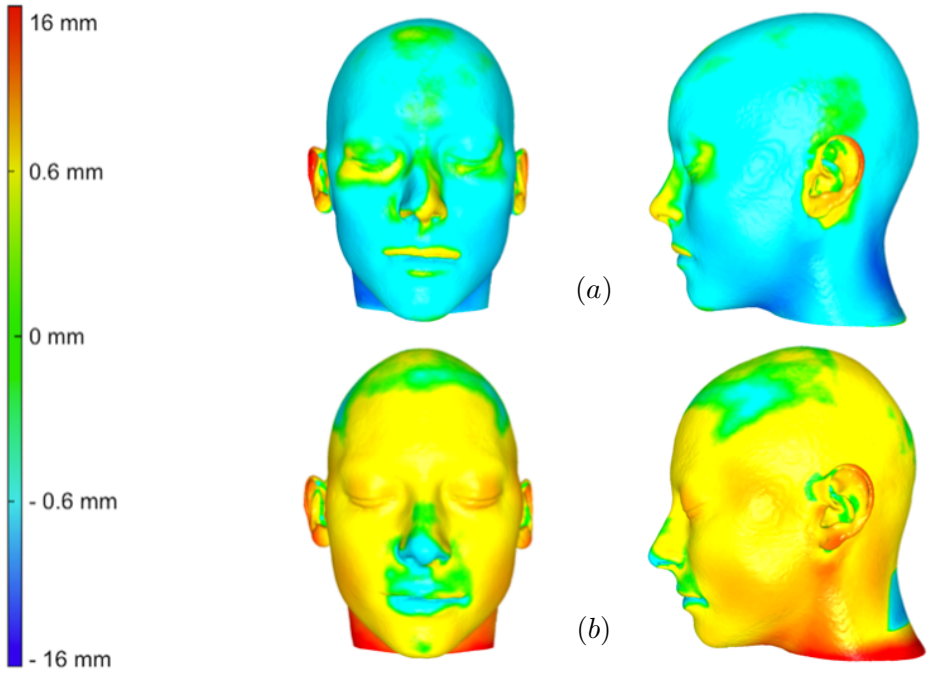


Figure 9: Discrepancy between the predicted and the original faces (a) for the test case of Figure 6; (b) for the test case of Figure 7.

experiments were carried out on a small toy collection of 26 individuals, our method produced promising results. Also, the combination of our approach with a more local one (using the reconstruction of facial muscles) seems to improve the accuracy of the overall method. In particular the proposed method for shape matching allows an accurate registration, opening for a large range of applications in computational anatomy. The method is simple to implement and doesn't need a prescribed landmarks correspondence. We hope to continue to discuss with the forensic and anthropology community for enriching our method and devising a validation protocol.

Appendix A

Existence results under additional constraints

In this appendix we provide a survey of classical results of existence in the context of shape optimization problems. These results are based on the idea of restraining the set of admissible shapes of a shape optimization problem by adding extra requirement of topological or geometrical nature, or by asking more regularity for the considered shapes.

A.0.1 Imposing a uniform cone property

A first idea exploited in [21] consist in adding a regularity constraint on the set of admissible shapes. More precisely we consider shapes enjoying the so-called uniform cone property, which is defined above.

Definition A.0.1. *Let $\xi \in \mathbb{R}^d$ be a unit vector, $y \in \mathbb{R}^d$ be a point and $h > 0, \theta \in (0, \frac{\pi}{2})$ be real numbers. The cone of angle θ, height h and axis ξ is the set:*

$$C(\theta, h, \xi) = \left\{ z \in \mathbb{R}^d : \langle z, \xi \rangle_{\mathbb{R}^d} > ||z|| \cos\theta, ||z|| < h \right\}.$$

Definition A.0.2. *Let $\theta \in (0, \frac{\pi}{2}), h > 0$ and $\varepsilon > 0$ real numbers, with $2\varepsilon \leq h$. An open set Ω is said to verify the uniform cone property if for every $x \in \partial\Omega$, there exists a unit vector ξ_x such that, for all point y belonging to $\Omega \cap B(x, \varepsilon)$, where $B(x, \varepsilon)$ is a ball of center x and ray ε, then one has:*

$$y + C(h, \theta, \xi_x) \subset \Omega, \qquad \forall y \in \Omega \cap B(x, \varepsilon).$$

Given a fixed design region $D \subset \mathbb{R}^d$, define the space:

$$\mathcal{O}_{\theta,h,r} = \{\Omega \subset D, \Omega \text{ has the uniform cone property for some } \theta, h, r\}.$$

The following result has been proved in [21].

Theorem A.0.1. *(Compactness of $\mathcal{O}_{\theta,h,r}$) Let $(\Omega_n)_{n\in\mathbb{N}}$ be a sequence in $\mathcal{O}_{\theta,h,r}$ converging to Ω^* in the sense of characteristic functions. Then $\Omega^* \in \mathcal{O}_{\theta,h,r}$.*

The above compactness property leads to existence results for a large class of objective functionals. The following results can be found in [1].

Theorem A.0.2. *The compliance minimization problem with a volume constraint:*

$$\min_{\Omega\in\mathcal{O}_{\theta,h,r,V}} \int_\Omega f u_\Omega(x)dx,$$

where u_Ω is the solution of (3.3.1) and $\mathcal{O}_{\theta,h,r,V} := \{\Omega \in \mathcal{O}_{\theta,h,r}, \int_D \chi_\Omega = V\}$, admits at least one solution.

A.0.2 The Murat-Simon regularity constraint

We are now going to present another regularity constraint due to Murat and Simon (see [81]) which is strongly linked with the notion of shape derivative that will be introduced in Section 1.2.1. Let us consider an open, bounded domain $\Omega_0 \subset \mathbb{R}^d$. Let us introduce the space of diffeomorphisms which are bi-Lipschitz maps:

$$\mathcal{T} := \{T : \mathbb{R}^d \to \mathbb{R}^d, (T - I) \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d) \text{ and } (T^{-1} - I) \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d)\}.$$

The main idea is to consider the space of admissible designs as the space of all domains obtained by applying a diffeomorphism of $\mathcal{T}$ on the reference domain Ω_0. Let us define the space of all the domains which can be represented as images of Ω_0 though an element of $\mathcal{T}$:

$$\mathcal{D}(\Omega_0) = \{\Omega \subset \mathbb{R}^d, \exists T \in \mathcal{T} \text{ such that } \Omega = T(\Omega_0)\}.$$

For Ω_1, Ω_2 we can define the following pseudo-distance:

$$d_{\mathcal{D}(\Omega_0)}(\Omega_1, \Omega_2) = \inf_{T\in\mathcal{T}, T(\Omega_1)=\Omega_2} (||T - I||_{W^{1,\infty}} + ||T^{-1} - I||_{W^{1,\infty}}).$$

Finally for each $C > 0$ let us define the space of admissible designs as the space

$$\mathcal{O}_{\Omega_0,C,V} = \{\Omega \in \mathcal{D}(\Omega_0), d_{\mathcal{D}(\Omega_0)}(\Omega_1, \Omega_2) \leq C, \int_\Omega dx = V\},$$

then we have the following existence result.

Theorem A.0.3. *The problem*

$$\min_{\Omega\in\mathcal{O}_{\Omega_0,C,V}} C(\Omega), \text{ where } C(\Omega) \text{ is defined in (3.3.2),} \tag{A.0.1}$$

has at least one solution.

A.0.3 Sets of finite perimeter

Many shape optimization problems involve constraints on the perimeter of the admissible sets. For example it is natural to impose the perimeter to be bounded to control the cost of the design. A theoretical important reason to add perimeter constraints in the optimization problems is due to the compactness property of sets of finite perimeter. We need to work with a generalized notion of perimeter.

Definition A.0.3. *(Perimeter in the sense of De Giorgi) Let $D \subset \mathbb{R}^d$ be an open set. Let $\Omega \subset D$ be a measurable set. The perimeter of Ω is defined by*

$$Per_D(\Omega) = \sup \left\{ \int_\Omega div\,(\phi)\,dx \qquad |\phi \in \mathcal{D}(D, \mathbb{R}^d), ||\phi||_\infty \leq 1 \right\},$$

where $\mathcal{D}(D, \mathbb{R}^d)$ is the space of C^∞ functions with compact support equipped with the norm:

$$||\phi||_\infty = \sup_{x \in D} |\phi(x)|.$$

The following proposition shows that the above definition coincides with the usual one for regular sets. The proof of this proposition can be found in [60].

Proposition A.0.1. *If Ω is of class C^1, then $Per_D(\Omega) = \int_{\partial\Omega} ds$, where ds is the surface element on $\partial\Omega$.*

Theorem A.0.4. *(Compactness property of sets of finite perimeter) Let $D \subset \mathbb{R}^d$ be a domain of finite perimeter. Let $(\Omega_n)_{n\in\mathbb{N}} \subset D$ be a sequence of measurable sets verifying*

$$|\Omega_n| + \ Per_D(\Omega_n) \leq C,$$

for some constant $C > 0$. Then there exist $\Omega \subset D$ of finite perimeter and a subsequence $(\Omega_{n_k})_{k\in\mathbb{N}}$ such that

$$\chi_{\Omega_{n_k}} \to \chi_\Omega \text{ in } L^1(D).$$

The above compactness property reposes on the following characterization of sets of finite perimeter.

Proposition A.0.2. *Let Ω be a measurable set in D. Then $Per_D(\Omega)$ is finite iff ∇_{χ_Ω} is a Radon measure of finite mass. Moreover in that case, $Per_D(\Omega) = ||\nabla_{\chi_\Omega}||_1$.*

A.0.4 Imposing a finite number of connected components

As for the perimeter, the number of connected components of the minimizing sequence $(\Omega_n)_{n\in\mathbb{N}}$ converging to a non classical design is unbounded. So, one can actually prevent the occurrence of homogenization phenomena by imposing an upper bound on the number of connected components of the admissible shapes. The following existence results due to Sverak [101] is valid only in two space dimension. A

similar result in linear elasticity setting has been provided in Chambolle [19]. For a given $k \in \mathbb{N}$ let us consider the design space

$$\mathcal{O}_k = \{\Omega \subset \mathbb{R}^d, \Omega \text{ has less then } k \text{ connected components }\},$$

then we have the following existence result.

Theorem A.0.5. *The problem*

$$\min_{\Omega \in \mathcal{O}_k} C(\Omega), \text{ where } C(\Omega) \text{ is defined in (3.3.2).} \tag{A.0.2}$$

admits at least one solution.

A.1 A non existence result using the topological gradient

In this section we show how to use the notion of topological derivative to get non existence of classical minimizers. We follow the approach described in [59]. When the functional at hand depends on a state u_Ω which is the solution of a PDE one has to inspect the behavior of the state $u_{\Omega,\rho}$ related to the perforated domain Ω_ρ in terms of the state u_Ω related to the reference domain Ω. Asymptotic expansions of such type has been studied in the context of identification of inhomogeneities in conductivity problems. In [45] an asymptotic formula is established for prooving the continuous dependence of the inverse problem. Let us introduce a design space $D \subset \mathbb{R}^2$. Let us consider a state $u_\Omega \in H^1(\Omega)$ solution of the following problem

$$\begin{cases} -\Delta u_\Omega = 1 & \text{in}\Omega, \\ u_\Omega = 0 & \text{on } \Gamma_D, \\ \nabla u_\Omega \cdot n = 0 & \text{on } \Gamma_N, \end{cases} \tag{A.1.1}$$

with $\Gamma_D, \Gamma_N \subset \partial\Omega$. Let us introduce the set of admissible shapes $\mathcal{O}_V$:

$$\mathcal{O}_V = \{\Omega \in D, \quad \Gamma_D \cap \Gamma_N \subset \partial\Omega, \quad |\Omega| = V\}.$$

Consider the following minimization problem:

$$\min_{\Omega \in \mathcal{O}_V} C(\Omega), \text{ where } C(\Omega) := \int_\Omega u_\Omega(x)dx. \tag{A.1.2}$$

The non existence result is the following.

Theorem A.1.1. *Problem* (A.1.2) *has no solution over* $\mathcal{O}_V$.

Proof. Let Ω^* be an optimal domain for (A.1.2). Let $x_0 \in \Omega^*$ a point and let $B(x_0, \varepsilon)$ a small circular hole centered in x_0. Since by perforating the domain we have created a new boundary we need to impose new boundary conditions on $\partial B(x_0, \varepsilon)$. Let us consider Dirichlet homogeneous condition. Thus the state u_{ε,x_0} corresponding

to the phase $\Omega^* \setminus B(x_0, \varepsilon)$ is the solution belonging to $H^1_{\Gamma_D}(\Omega^* \setminus B(x_0, \varepsilon))$ of the problem:

$$\begin{cases} -\Delta u_{\varepsilon,x_0} = 1 & \text{in } \Omega^* \setminus B_{\varepsilon,x_0} \\ u_{\varepsilon,x_0} = 0 & \text{on } \Gamma_D \\ u_{\varepsilon,x_0} = 0 & \text{on } \partial B_{\varepsilon,x_0} \\ \nabla u_{\varepsilon,x_0} \cdot n = 0 & \text{on } \Gamma_N \end{cases} \tag{A.1.3}$$

The following asymptotic expansion holds (see [59]):

$$u_{\varepsilon,x_0}(y) = u(y) - \frac{2\pi}{|\log \varepsilon|} u(x)\phi_x(y) + o\left(\frac{1}{|\log \varepsilon|}\right). \tag{A.1.4}$$

where $\phi_x(y)$ is the Green function of the Laplace operator vanishing on $\partial\Omega$. In order to respect the volume constraint we consider a point x_1 lying in the complementary $D \setminus \Omega^*$ and we add to the domain Ω^* a circular region of ray ε centered in x_1. We define

$$\Omega^*_\varepsilon := (\Omega^* \setminus B(x_0, \varepsilon)) \cup B(x_1, \varepsilon),$$

and we investigate the sign of the difference $J(\Omega^*_\varepsilon) - J(\Omega^*)$. Since Ω^*_ε is composed of two disjoints components we can evaluate the state $u_\varepsilon := u_{\Omega^*_\varepsilon}$ separately on $\Omega^* \setminus B(x_0, \varepsilon)$ and $B(x_1, \varepsilon)$:

$$u_{\Omega^*_\varepsilon}(y) = \begin{cases} u_{\Omega^* \setminus B(x_0,\varepsilon)}(y) & \text{if } y \in \Omega^* \setminus B(x_0, \varepsilon), \\ \dfrac{\varepsilon^2 - |y - x_1|^2}{4} & \text{if } y \in B(x_1, \varepsilon). \end{cases} \tag{A.1.5}$$

We can now evaluate the shape difference:

$$\begin{aligned} J(\Omega^*_\varepsilon) &= \int_{\Omega \setminus B(x_0,\varepsilon)} u_\varepsilon(y)dy + \int_{B(x_1,\varepsilon)} u_\varepsilon(y)dy \\ &= \int_\Omega u_\varepsilon(y)dy + \int_{B_{\varepsilon,x_1}} u_\varepsilon(y)dy \\ &= \int_\Omega u(y)dy - \frac{2\pi}{|\log \varepsilon|} u(x_0) \int_\Omega \phi_{x_0}(y)dy + \int_{B(x_1,\varepsilon)} \frac{\varepsilon^2 - |y - x_1|^2}{4} dy \\ &= J(\Omega) - \frac{2\pi}{|\log \varepsilon|} u(x_0) \int_\Omega \phi_{x_0}(y)dy + \int_{B(x_1,\varepsilon)} \frac{\varepsilon^2 - |y - x_1|^2}{4} dy + o\left(\frac{1}{|\log \varepsilon|}\right) \\ &= J(\Omega) - \frac{2\pi}{|\log \varepsilon|} u(x_0) \int_\Omega \phi_{x_0}(y)dy + o\left(\frac{1}{|\log \varepsilon|}\right). \end{aligned}$$

Remarking that by the maximum principle $u(x_0) > 0$ and $\phi_{x_0}(y) > 0$ for y close enough to x_0 we end up with

$$J(\Omega_\varepsilon(y)) < J(\Omega) \text{ for } y \text{ close to } x_0,$$

which contradicts the optimality of Ω.

Appendix B

Strain/displacement interpolation for shell Finite Elements

B.0.1 Strain/displacement interpolation

In this appendix we report the computation of the derivatives of the discrete displacement in context of shell Finite Elements introduced in Chapter 4. We refer to the notations introduced in 4. The Jacobian matrix of the mapping $(r, s, t) \to (x, y, z)$ is easily calculated as:

$$J = \begin{bmatrix} x^{(0)} - x^{(2)} + t\frac{a}{2}(n_x^0 - n_x^2) & y^{(0)} - y^{(2)} + t\frac{a}{2}(n_y^0 - n_y^2) & z^{(0)} - z^{(2)} + t\frac{a}{2}(n_z^0 - n_z^2) \\ x^{(1)} - x^{(2)} + t\frac{a}{2}(n_x^1 - n_x^2) & y^{(1)} - y^{(2)} + t\frac{a}{2}(n_y^1 - n_y^2) & z^{(1)} - z^{(2)} + t\frac{a}{2}(n_z^1 - n_z^2) \\ \sum_{i=0}^{3} h_i(r,s)n_x^i & \sum_{i=0}^{3} h_i(r,s)n_y^i & \sum_{i=0}^{3} h_i(r,s)n_z^i \end{bmatrix}$$

Mathematical shell models are classically based on the so called Reissner-Mindilin kinematical assumption : any material line orthogonal to the midsurface in the unstressed configuration remains straight and unstretched during deformations. Such a assumptions allows for the global deplacement field written as the sum of two contributes : the displacement of the shell midsurface and a rotation field encoding the angle between the unit normal vector in the unstressed and deformed configuration. The resulting finite element model shows 15 degree of freedom on each element (9 for midsurface displacement and 6 for angles). The interpolated displacement inside the element T_k^e is then written under the form

$$u(r,s,t) = \sum_{i=0}^{3} h_i(r,s)\big(u^{(i)} + t\frac{a}{2}V_x^i\big), \tag{B.0.1}$$

$$v(r,s,t) = \sum_{i=0}^{3} h_i(r,s)\big(v^{(i)} + t\frac{a}{2}V_y^i\big), \tag{B.0.2}$$

$$w(r,s,t) = \sum_{i=0}^{3} h_i(r,s)\big(w^{(i)} + t\frac{a}{2}V_z^i\big). \tag{B.0.3}$$

The coordinates of the updated normals V^i at each node are computed w.r.t. an orthonormal basis $\{V_0^i, V_1^i, V_2^i\}$. A simple choice for the basis vectors is

$$V_0^i = N^i, \tag{B.0.4}$$

$$V_1^i = \frac{a_y \times V_0^i}{||a_y \times V_0^i||}, \tag{B.0.5}$$

$$V_2^i = \frac{V_0 \times V_1^i}{||V_0 \times V_1^i||}. \tag{B.0.6}$$

for $i = 0, 1, 2$. The updated normal is calculated as

$$V^i = -\alpha_i V_2^i + \beta_i V_1^i,$$

where α_i (resp. β_i) are the rotation angles of N^i around V_1^i (resp. V_2^i).

The (B.0.1) turns into

$$u(r,s,t) = \sum_{i=0}^{3} h_i(r,s)\big(u^{(i)} + t(\alpha_i g_{1x}^i + \beta_i g_{2x}^i)\big) \tag{B.0.7}$$

$$v(r,s,t) = \sum_{i=0}^{3} h_i(r,s)\big(v^{(i)} + t(\alpha_i g_{1y}^i + \beta_i g_{2y}^i)\big) \tag{B.0.8}$$

$$w(r,s,t) = \sum_{i=0}^{3} h_i(r,s)\big(w^{(i)} + t(\alpha_i g_{1z}^i + \beta_i g_{2z}^i)\big) \tag{B.0.9}$$

where

$$g_1^i = -\frac{a}{2}V_2^i \quad \text{and} \quad g_2^i = \frac{a}{2}V_1^i.$$

The displacement derivatives w.r.t. the reference coordinates (r, s, t) are written as:

$$\begin{bmatrix} \frac{\partial u}{\partial r} \\ \frac{\partial u}{\partial s} \\ \frac{\partial u}{\partial t} \\ \frac{\partial v}{\partial r} \\ \frac{\partial v}{\partial s} \\ \frac{\partial v}{\partial t} \\ \frac{\partial w}{\partial r} \\ \frac{\partial w}{\partial s} \\ \frac{\partial w}{\partial t} \end{bmatrix} = B \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ v_1 \\ v_2 \\ v_3 \\ w_1 \\ w_2 \\ w_3 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix},$$

where $B \in \mathcal{M}^{9\times 15}(\mathbb{R})$ is the block matrix

$$B = \begin{bmatrix} B_h & \cdots 0 \cdots & \cdots 0 \cdots & B_{g1x} & B_{g2x} \\ \cdots 0 \cdots & B_h & \cdots 0 \cdots & B_{g1y} & B_{g2y} \\ \cdots 0 \cdots & \cdots 0 \cdots & B_h & B_{g1z} & B_{g2z} \end{bmatrix}.$$

The matrices B_h, $\{B_{g1}\}_{x,y,z}, \{B_{g2}\}_{x,y,z} \in \mathcal{M}^{3\times 5}(\mathbb{R})$ are defined as

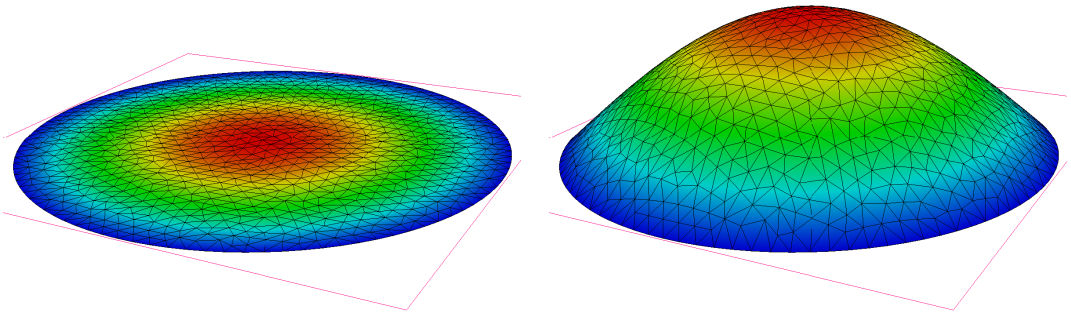
$$B_h = \begin{bmatrix} \frac{\partial h_1}{\partial r} & \frac{\partial h_2}{\partial r} & \frac{\partial h_3}{\partial r} \\ \frac{\partial h_1}{\partial s} & \frac{\partial h_2}{\partial s} & \frac{\partial h_3}{\partial s} \\ 0 & 0 & 0 \end{bmatrix},$$

$$\{B_{gi}\}_{x,y,z} = \begin{bmatrix} t\frac{\partial h_1}{\partial r} g_i^1 & t\frac{\partial h_2}{\partial r} g_i^2 & t\frac{\partial h_3}{\partial r} g_i^3 \\ t\frac{\partial h_1}{\partial s} g_i^1 & t\frac{\partial h_2}{\partial s} g_i^2 & t\frac{\partial h_3}{\partial s} g_i^3 \\ h_1 g_i^1 & h_2 g_i^2 & h_3 g_i^3 \end{bmatrix}_{x,y,z} \qquad i = 1, 2.$$

The displacement derivatives w.r.t. to the cartesian coordinates (x, y, z) are computed as in a standard finite element setting:

$$\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} \end{bmatrix} = J^{-1} \begin{bmatrix} \frac{\partial u}{\partial r} & \frac{\partial v}{\partial r} & \frac{\partial w}{\partial r} \\ \frac{\partial u}{\partial s} & \frac{\partial v}{\partial s} & \frac{\partial w}{\partial s} \\ \frac{\partial u}{\partial t} & \frac{\partial v}{\partial t} & \frac{\partial w}{\partial t} \end{bmatrix}.$$

Figure 1 depicts the deformation of a circular membrane which is clamped at its boundary ad submitted to a constant body force which is orthogonal to the midsuface.



(a) (b)

Figure 1: (a) Undeformed membrane. (b) Deformed membrane.

List of Figures

Bibliography

[1] Grégoire Allaire. Conception optimale de structures, volume 58 of mathématiques & applications (berlin)[mathematics & applications], 2007.

[2] Grégoire Allaire and Antoine Henrot. On some recent advances in shape optimization. *Comptes Rendus de l'Académie des Sciences - Series IIB - Mechanics*, 329(5):383 – 396, 2001.

[3] Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. Anisotropic polygonal remeshing. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 485–493. ACM, 2003.

[4] Pierre Alliez, EC De Verdire, Olivier Devillers, and Martin Isenburg. Isotropic surface remeshing. In *Shape Modeling International, 2003*, pages 49–58. IEEE, 2003.

[5] Pierre Alliez, Giuliana Ucelli, Craig Gotsman, and Marco Attene. Recent advances in remeshing of surfaces. *Shape analysis and structuring*, pages 53–82, 2008.

[6] Luigi Ambrosio. Lecture notes on geometric evolution problems, distance function and viscosity solutions. Technical report, Paris, France, France, 1997.

[7] Ruzena Bajcsy and Stane Kovačič. Multiresolution elastic matching. *Comput. Vision Graph. Image Process.*, 46(1):1–21, April 1989.

[8] Timothy J. Baker. Mesh movement and metamorphosis. In *in Proceedings of the Tenth International Meshing Roundtable, Albuquerque, NM, 2001, Sandia National Laboratories*, pages 387–396, 2001.

[9] Klaus-Jürgen Bathe. *Finite element procedures*. Klaus-Jurgen Bathe, 2006.

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

[11] Mirza Faisal Beg. *Variational and computational methods for flows of diffeomorphisms in image matching and growth in computational anatomy*. PhD thesis, Johns Hopkins University, 2004.

[12] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE transactions on pattern analysis and machine intelligence*, 24(4):509–522, 2002.

[13] Maxime Berar, Francoise M Tilotta, Joann A Glaunés, and Yves Rozenholc. Craniofacial reconstruction as a prediction problem using a latent root regression model. *Forensic science international*, 210(1):228–236, 2011.

[14] Marshall Bern and David Eppstein. Mesh generation and optimal triangulation. *Computing in Euclidean geometry*, 1:23–90, 1992.

[15] Fred L Bookstein. *Morphometric tools for landmark data: geometry and biology*. Cambridge University Press, 1991.

[16] H Borouchaki and PJ Frey. Simplification of surface mesh using hausdorff envelope. *Computer methods in applied mechanics and engineering*, 194(48):4864–4884, 2005.

[17] Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, and Bruno Lévy. *Polygon mesh processing*. CRC press, 2010.

[18] Jean Céa. Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 20(3):371–402, 1986.

[19] Antonin Chambolle and Christopher J Larsen. regularity of the free boundary for a two-dimensional optimal compliance problem. *Calculus of Variations and Partial Differential Equations*, 18(1):77–94, 2003.

[20] Dominique Chapelle and Klaus-Jurgen Bathe. *The finite element analysis of shells-fundamentals*. Springer Science & Business Media, 2010.

[21] Denise Chenais. On the existence of a solution in a domain identification problem. *Journal of Mathematical Analysis and Applications*, 52(2):189 – 219, 1975.

[22] P. G. Ciarlet. *Mathematical Elasticity: Theory of Shells*. Number v. 3 in Mathematical Elasticity. North-Holland, 2000.

[23] P.G. Ciarlet. *Mathematical Elasticity: Three-dimensional elasticity*. Number v. 1 in Mathematical Elasticity. North-Holland, 1994.

[24] Peter Claes. *A robust statistical surface registration framework using implicit function representations-application in craniofacial reconstruction*. PhD thesis, Katholieke Universiteit Leuven, 2007.

[25] Peter Claes, Dirk Vandermeulen, Sven De Greef, Guy Willems, John Gerald Clement, and Paul Suetens. Computerized craniofacial reconstruction: conceptual framework and review. *Forensic science international*, 201(1):138–145, 2010.

[26] Peter Claes, Dirk Vandermeulen, Sven De Greef, Guy Willems, John Gerald Clement, and Paul Suetens. Computerized craniofacial reconstruction: Conceptual framework and review. *Forensic Science International*, 201(1â3):138–145, 2010. Meeting of the International Organization of Forensic Odonto-Stomatology.

[27] Peter Claes, Dirk Vandermeulen, Sven De Greef, Guy Willems, and Paul Suetens. Craniofacial reconstruction using a combined statistical model of face shape and soft tissue depths: Methodology and validation. *Forensic Science International*, 159, Supplement:S147 – S158, 2006. International {IOFOS} Symposium on Forensic Odontology 2006 and 3rd International Conference on Reconstruciton of Soft Facial Parts 2006.

[28] A. Claisse and P. Frey. A nonlinear {PDE} model for reconstructing a regular surface from sampled data using a level set formulation on triangular meshes. *Journal of Computational Physics*, 230(12):4636 – 4656, 2011.

[29] Michael G Crandall, Hitoshi Ishii, and Pierre-Louis Lions. User's guide to viscosity solutions of second order partial differential equations. *Bulletin of the American Mathematical Society*, 27(1):1–67, 1992.

[30] Charles Dapogny. *Shape optimization, level set methods on unstructured meshes and mesh evolution.* PhD thesis, Université Pierre et Marie Curie-Paris 6, 2013.

[31] Charles Dapogny, Cécile Dobrzynski, and Pascal Frey. Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics*, 262:358–378, 2014.

[32] Charles Dapogny, Alexis Faure, Georgios Michailidis, Grégoire Allaire, Agnes Couvelas, and Rafael Estevez. Geometric constraints for shape and topology optimization in architectural design. *Computational Mechanics*, 59(6):933–965, 2017.

[33] Charles Dapogny and Pascal Frey. Computation of the signed distance function to a discrete contour on adapted triangulation. *Calcolo*, 49(3):193–219, 2012.

[34] Charles Dapogny, Pascal Frey, Florian Omnès, and Yannick Privat. Geometrical shape optimization in fluid mechanics using freefem++. 2017.

[35] Frédéric De Gournay. Velocity extension for the level-set method and multiple eigenvalues in shape optimization. *SIAM journal on control and optimization*, 45(1):343–367, 2006.

[36] Sven De Greef, Peter Claes, Dirk Vandermeulen, Wouter Mollemans, Paul Suetens, and Guy Willems. Large-scale in-vivo caucasian facial soft tissue thickness database for craniofacial reconstruction. *Forensic science international*, 159:S126–S146, 2006.

[37] Michel Desvignes, Gerard Bailly, Yohan Payan, and Maxime Berar. 3d semi-landmarks based statistical face reconstruction. *CIT. Journal of computing and information technology*, 14(1):31–43, 2006.

[38] Akio Doi and Akio Koide. An efficient method of triangulating equi-valued surfaces by using tetrahedral cells. *IEICE TRANSACTIONS on Information and Systems*, 74(1):214–224, 1991.

[39] Ye Duan, Liu Yang, Hong Qin, and Dimitris Samaras. Shape reconstruction from 3d and 2d data using pde-based deformable surfaces. *Computer Vision-ECCV 2004*, pages 238–251, 2004.

[40] Paul Dupuis, Ulf Grenander, and Michael I Miller. Variational problems on flows of diffeomorphisms for image matching. *Quarterly of applied mathematics*, pages 587–600, 1998.

[41] WD Evans. Partial differential equations, 1988.

[42] Michael S Floater and Kai Hormann. Surface parameterization: a tutorial and survey. In *Advances in multiresolution for geometric modelling*, pages 157–186. Springer, 2005.

[43] D Franklin, L Freedman, and N Milne. Sexual dimorphism and discriminant function sexing in indigenous south african crania. *HOMO-Journal of Comparative Human Biology*, 55(3):213–228, 2005.

[44] Pascal Jean Frey and Paul L George. *Mesh generation: application to finite elements*. Wiley Online Library, 2000.

[45] Avner Friedman and Michael Vogelius. Identification of small inhomogeneities of extreme conductivity by boundary measurements: a theorem on continuous dependence. *Archive for Rational Mechanics and Analysis*, 105(4):299–326, 1989.

[46] Mikhail Gerasimov et al. Face finder. 1971.

[47] MM Gerasimov. The reconstruction of the face from the basic structure of the skull. *Russia: Publisher Unknown*, 1955.

[48] Eugene Giles and Orville Elliot. Sex determination by discriminant function analysis of crania. *American Journal of Physical Anthropology*, 21(1):53–68, 1963.

[49] Joan Glaunès. *Transport par difféomorphismes de points, de mesures et de courants pour la comparaison de formes et l?anatomie numérique*. PhD thesis, These de sciences, Université Paris, 2005.

[50] Joan Glaunès, Anqi Qiu, Michael I Miller, and Laurent Younes. Large deformation diffeomorphic metric curve mapping. *International journal of computer vision*, 80(3):317–336, 2008.

[51] Joan Glaunes, Alain Trouvé, and Laurent Younes. Diffeomorphic matching of distributions: A new approach for unlabelled point-sets and sub-manifolds matching. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–II. Ieee, 2004.

[52] Joan Glaunès, Alain Trouvé, and Laurent Younes. Modeling planar shape variation via hamiltonian flows of curves. *Statistics and analysis of shapes*, pages 335–361, 2006.

[53] Joan Glaunès, Marc Vaillant, and Michael I Miller. Landmark matching via large deformation diffeomorphisms on the sphere. *Journal of mathematical imaging and vision*, 20(1-2):179–200, 2004.

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

[55] Carl Martin Grewe, Lisa Schreiber, and Stefan Zachow. Fast and accurate digital morphometry of facial expressions. *Facial Plastic Surgery*, 31(05):431–438, 2015.

[56] Xianfeng Gu and Baba C Vemuri. Matching 3d shapes using 2d conformal representations. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 771–780. Springer, 2004.

[57] William D Haglund and Donald T Reay. Use of facial approximation techniques in identification of green river serial murder victims. *The American journal of forensic medicine and pathology*, 12(2):132–142, 1991.

[58] Steven Haker, Lei Zhu, Allen Tannenbaum, and Sigurd Angenent. Optimal mass transport for registration and warping. *International Journal of computer vision*, 60(3):225–240, 2004.

[59] Antoine Henrot, Ilaria Lucardesi, and Gérard Philippin. On two functionals involving the maximum of the torsion function, 2017.

[60] Antoine Henrot and Michel Pierre. *Variation et optimisation de forme*. Springer, 2005.

[61] Masaki Hilaga, Yoshihisa Shinagawa, Taku Kohmura, and Tosiyasu L Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 203–212. ACM, 2001.

[62] Maria Isabel Huete, Oscar Ibáñez, Caroline Wilkinson, and Tzipi Kahana. Past, present, and future of craniofacial superimposition: literature and international surveys. *Legal Medicine*, 17(4):267–278, 2015.

[63] Hyeon-Shik Hwang, Myoung-Kyu Park, Won-Joon Lee, Jin-Hyoung Cho, Byung-Kuk Kim, and Caroline M Wilkinson. Facial soft tissue thickness database for craniofacial reconstruction in korean adults. *Journal of forensic sciences*, 57(6):1442–1447, 2012.

[64] Mark W Jones. Facial reconstruction using volumetric data. In *VMV*, pages 135–150. Citeseer, 2001.

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

[66] Leif P Kobbelt, Mario Botsch, Ulrich Schwanecke, and Hans-Peter Seidel. Feature sensitive surface extraction from volume data. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 57–66. ACM, 2001.

[67] Wilton Marion Krogman. The human skeleton in forensic medicine. 1962.

[68] Agnes Kustár, Laszlo Forró, Ildiko Kalina, Ferenc Fazekas, Szabolcs Honti, Szabolcs Makra, and Martin Friess. Facer - a $3d$ database of 400 living individuals full head ct-and face scans and preliminary gmm analysis for craniofacial reconstruction. *Journal of forensic sciences*, 58(6):1420–1428, 2013.

[69] Hervé Le Dret. Méthodes mathématiques en élasticité. *Notes de cours de DEA*, 2004, 2003.

[70] Bruno Lévy. A numerical algorithm for l2 semi-discrete optimal transport in 3d. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6):1693–1715, 2015.

[71] Thibault Liard, Pierre Lissy, and Yannick Privat. Non-localization of eigenfunctions for sturm-liouville operators. 2015.

[72] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *ACM siggraph computer graphics*, volume 21, pages 163–169. ACM, 1987.

[73] Lok Ming Lui, Sheshadri Thiruvenkadam, Yalin Wang, Paul M Thompson, and Tony F Chan. Optimized conformal surface registration with shape-based landmark matching. *SIAM Journal on Imaging Sciences*, 3(1):52–78, 2010.

[74] P do Carmo Manfredo. Differential geometry of curves and surfaces, 1976.

[75] Sergey V Matveyev. Approximation of isosurface in the marching cube: Ambiguity problem. In *Visualization, 1994., Visualization'94, Proceedings., IEEE Conference on*, pages 288–292. IEEE, 1994.

[76] Facundo Mémoli. On the use of gromov-hausdorff distances for shape comparison. 2007.

[77] Mark Meyer, Mathieu Desbrun, Peter Schröder, Alan H Barr, et al. Discrete differential-geometry operators for triangulated 2-manifolds. *Visualization and mathematics*, 3(2):52–58, 2002.

[78] James V Miller, David E Breen, William E Lorensen, Robert M O'Bara, and Michael J Wozny. Geometrically deformed models: a method for extracting closed geometric models form volume data. In *ACM SIGGRAPH Computer Graphics*, volume 25, pages 217–226. ACM, 1991.

[79] Sachio Miyasaka, M Yoshino, K Imaizumi, and S Seta. The computer-aided facial reconstruction system. *Forensic science international*, 74(1-2):155–165, 1995.

[80] Claudio Montani, Riccardo Scateni, and Roberto Scopigno. A modified look-up table for implicit disambiguation of marching cubes. *The Visual Computer*, 10(6):353–355, 1994.

[81] Francois Murat and Jacques Simon. *Quelques résultats sur le contrôle par un domaine géométrique*. VI Laboratoire d'Analyse Numérique, 1974.

[82] Paul Mansour Naghdi. The theory of shells and plates. In *Linear Theories of Elasticity and Thermoelasticity*, pages 425–640. Springer, 1973.

[83] Kamal Nasreddine, Abdesslam Benzinou, and Ronan Fablet. Variational shape matching for shape classification and retrieval. *Pattern Recognition Letters*, 31(12):1650–1657, 2010.

[84] LA Nelson and SD Michael. The application of volume deformation to three-dimensional facial reconstruction: a comparison with previous techniques. *Forensic Science International*, 94(3):167–181, 1998.

[85] Gregory M Nielson and Bernd Hamann. The asymptotic decider: resolving the ambiguity in marching cubes. In *Proceedings of the 2nd conference on Visualization'91*, pages 83–91. IEEE Computer Society Press, 1991.

[86] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[87] Pascal Paysan, Marcel Lüthi, Thomas Albrecht, Anita Lerch, Brian Amberg, Francesco Santini, and Thomas Vetter. Face reconstruction from skull shapes and physical attributes. In *Joint Pattern Recognition Symposium*, pages 232–241. Springer, 2009.

[88] Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental mathematics*, 2(1):15–36, 1993.

[89] Benjamin J Pomidor, Jana Makedonska, and Dennis E Slice. A landmark-free method for three-dimensional shape analysis. *PloS one*, 11(3):e0150368, 2016.

[90] John Prag and Richard Neave. Making faces: using forensic and archaeological evidence. 1997.

[91] Gérald Quatrehomme, Stéphane Cotin, Gérard Subsol, Hervé Delingette, Yves Garidel, Gilles Grévin, Martha Fidrich, Paul Bailet, and Amédée Ollier. A fully three-dimensional method for facial reconstruction based on deformable models. *Journal of Forensic Science*, 42(4):649–652, 1997.

[92] J-F Remacle, Christophe Geuzaine, G Compere, and E Marchandise. High-quality surface remeshing using harmonic maps. *International Journal for Numerical Methods in Engineering*, 83(4):403–425, 2010.

[93] Stefan Schlager. *Soft-tissue reconstruction of the human nose: population differences and sexual dimorphism.* PhD thesis, Dissertation, Albert-Ludwigs-Universität Freiburg, 2013, 2013.

[94] James A Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.

[95] James A Sethian. Fast marching methods. *SIAM review*, 41(2):199–235, 1999.

[96] Konrad Simon and Ronen Basri. Elasticity-based matching by minimizing the symmetric difference of shapes. *arXiv preprint arXiv:1510.04563*, 2015.

[97] Konrad Simon, Sameer Sheorey, David Jacobs, and Ronen Basri. A linear elastic force optimization model for shape matching. *Journal of Mathematical Imaging and Vision*, 51(2):260–278, 2015.

[98] Detlev Stalling, Malte Westerhoff, and Hans-Christian Hege. 38–amira: a highly interactive system for visual data analysis. *Visualization Handbook*, 2005.

[99] Zhengyu Su, Yalin Wang, Rui Shi, Wei Zeng, Jian Sun, Feng Luo, and Xianfeng Gu. Optimal mass transport for shape matching and comparison. *IEEE transactions on pattern analysis and machine intelligence*, 37(11):2246–2259, 2015.

[100] Paul Suetens, Guy Willems, Dirk Vandermeulen, Sven De Greef, and Peter Claes. Statistically deformable face models for cranio-facial reconstruction. *CIT. Journal of computing and information technology*, 14(1):21–30, 2006.

[101] A Sverak. On existence of solution for a class of optimal shape design problems. *Note CRAS*, 34:92–0025, 1992.

[102] Karen T Taylor. *Forensic art and illustration.* CRC Press, 2000.

[103] Francoise Tilotta. *Contribution à la reconstitution faciale en médecine légale: proposition d'une nouvelle méthode statistique.* PhD thesis, These de sciences, Université Paris, 2008.

[104] Francoise Tilotta, Frédéric Richard, Joan Glaunès, Maxime Berar, Servane Gey, Stéphane Verdeille, Yves Rozenholc, and Jean-Francois Gaudy. Construction and analysis of a head ct-scan database for craniofacial reconstruction. *Forensic science international*, 191(1):112–e1, 2009.

[105] Francoise Tilotta, Frédéric Richard, Joan Glaunès, Maxime Berar, Servane Gey, Stéphane Verdeille, Yves Rozenholc, and Jean-Francois Gaudy. Construction and analysis of a head ct-scan database for craniofacial reconstruction. *Forensic science international*, 191(1):112–e1, 2009.

[106] Peter Tu, Rebecca Book, Xiaoming Liu, Nils Krahnstoever, Carl Adrian, and Phil Williams. Automatic face recognition from skeletal remains. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–7. IEEE, 2007.

[107] Peter Tu, Richard Hartley, William E Lorensen, Majeid Allyassin, Rajiv Gupta, Linda Heier, et al. *Face reconstruction using flesh deformation modes*. Elsevier Academic Press, 2005.

[108] Marc Vaillant and Joan Glaunès. Surface matching via currents. In *Information processing in medical imaging*, pages 1–5. Springer, 2005.

[109] Dirk Vandermeulen, Peter Claes, Dirk Loeckx, Sven De Greef, Guy Willems, and Paul Suetens. Computerized craniofacial reconstruction using ct-derived implicit surface representations. *Forensic science international*, 159:S164–S174, 2006.

[110] P Vanezis, RW Blowes, AD Linney, AC Tan, R Richards, and R Neave. Application of 3-d computer graphics for facial reconstruction and comparison with sculpting techniques. *Forensic science international*, 42(1-2):69–84, 1989.

[111] Remco C Veltkamp. Shape matching: Similarity measures and algorithms. In *Shape Modeling and Applications, SMI 2001 International Conference on.*, pages 188–197. IEEE, 2001.

[112] Alex Vlachos, Jörg Peters, Chas Boyd, and Jason L Mitchell. Curved pn triangles. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 159–166. ACM, 2001.

[113] Sen Wang, Yang Wang, Miao Jin, Xianfeng Gu, and Dimitris Samaras. 3d surface matching and recognition using conformal geometry. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2453–2460. IEEE, 2006.

[114] Hermann Welcker. On the skull of dante. *The Anthropological Review*, 5(16):56–71, 1867.

[115] Rüdiger Westermann, Leif Kobbelt, and Thomas Ertl. Real-time exploration of regular volume data by adaptive reconstruction of isosurfaces. *The Visual Computer*, 15(2):100–111, 1999.

[116] Tim D White and Pieter A Folkens. *The human bone manual*. Academic Press, 2005.

[117] Jane Wilhelms and Allen Van Gelder. Octrees for faster isosurface generation. *ACM Transactions on Graphics (TOG)*, 11(3):201–227, 1992.

[118] Caroline Wilkinson. Computerized forensic facial reconstruction. *Forensic Science, Medicine, and Pathology*, 1(3):173–177, 2005.

[119] Caroline Wilkinson. Facial reconstruction–anatomical art or artistic anatomy? *Journal of anatomy*, 216(2):235–250, 2010.

[120] Miriam Leah Zelditch, Donald L Swiderski, and H David Sheets. *Geometric morphometrics for biologists: a primer*. Academic Press, 2012.

[121] Hong-Kai Zhao, Stanley Osher, and Ronald Fedkiw. Fast surface reconstruction using the level set method. In *Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on*, pages 194–201. IEEE, 2001.

[122] Hongkai Zhao. A fast sweeping method for eikonal equations. *Mathematics of computation*, 74(250):603–627, 2005.

Mesh deformation strategies in shape optimization
Application to forensic facial reconstruction

Abstract :
This thesis is devoted to the conception, the development and the analysis of mesh deformation strategies for shape modeling, processing and comparison -as shape matching and surface reconstruction- and, in a rather independent concern, for devising a robust computational method for facial reconstruction. Facial reconstruction is about the estimation of a facial shape from the sole datum of the underlying skull and is a challenging problem in anthropology and forensic science. The main contribution of the thesis is the design of a novel method for shape matching, borrowing techniques from the shape optimization context. Under the sole assumption that the two shapes share the same topology, the desired mapping is achieved as a sequence of elastic displacements by minimizing an energy functional based on a signed distance function. Several numerical examples are presented to show the efficiency of the method. Also, a novel method for generating a closed surface mesh model of an initially non-closed source mesh model is developed. The method relies on an original PDE-based mesh evolution technique. A template shape is iteratively deformed, producing a sequence of shapes that get 'closer and closer' to the source triangulation. The second part of the manuscript deals with the development of a landmark-free, fully automated method for digital facial reconstruction. Based on techniques of continuous deformation as 'morphing' and 'warping', the proposed approach is integrated with anthropological assumptions and mechanical models.
Keywords : shape matching, shape optimization, liner elasticity, mesh deformation, finite elements, facial reconstruction

Méthodes de déformation de maillage en optimisation de forme
Application à la reconstruction faciale pour la médecine légale

Résumé :
Cette thèse est consacrée à la conception, au développement et à l'analyse de méthodes de déformation de maillage pour la modélisation, le traitement et la comparaison de forme -telles que l'appariement et la reconstruction de surface- ainsi qu'à la conception d'une méthode numérique robuste pour la reconstruction faciale. La reconstruction faciale tridimensionnelle consiste à estimer un visage numérique à partir de la seule donnée de son crâne sec. Il s'agit d'un défi en médecine légale et en anthropologie. La contribution majeure de cette thèse est la conception d'une nouvelle méthode pour l'appariement de forme, en s'appuyant sur des techniques d'optimisation de forme. Sous la seule hypothèse que les deux formes ont la même topologie, la transformation cherchée s'obtient comme une suite de déplacements élastiques, solutions d'un problème de minimisation d'énergie basée sur une fonction distance signée. Nous proposons également une méthode de drapage permettant la génération d'un modèle de surface fermée à partir d'un maillage source. La méthode repose sur une technique d'évolution de maillage utilisant les équations de l'élasticité linéaire. Un maillage modèle est itérativement déformé pour générer une séquence de formes qui s'approche de plus en plus de la triangulation source. Dans la deuxième partie de ce manuscrit, nous nous intéressons à une méthode automatique de reconstruction faciale numérique. En s'appuyant sur des techniques de déformation continue telles que le 'morphing' et le 'warping', l'approche proposée est intégrée par des connaissances anthropologiques et mécaniques.
Mot clés : appariement de forme, optimisation de forme, élasticité linéaire, déformation de maillage, éléments finis, reconstruction faciale