An optimization method for elastic shape matching

Maya de Buhan, Charles Dapogny, Pascal Frey, Chiara Nardoni

An optimization method for elastic shape matching

Maya de Buhan [a], Charles Dapogny [b] , Pascal Frey [c] , Chiara Nardoni [c,1]

[a] *MAP5, CNRS UMR 8145, Université Paris Descartes, Sorbonne Paris Cité, France*

[b] *Laboratoire Jean Kuntzmann, CNRS, Université Joseph Fourier, Grenoble INP, Université Pierre Mendés France, BP 53, 38041 Grenoble Cedex 9, France*

[c] *Sorbonne Universités, UPMC Univ Paris* 06*, Institut du calcul et de la simulation (ICS), F-75005, Paris, France*

Received *****; accepted after revision +++++

Presented by

Abstract

This note addresses the following shape matching problem: given a 'template' shape, numerically described by means of a computational mesh, and a 'target' shape, known only via a signed distance function to its boundary, we aim at deforming iteratively the mesh of the template shape into a computational mesh of the target shape. To achieve this goal, we rely on techniques from shape optimization. Under the sole assumption that both shapes share the same topology, the desired transformation is realized as a sequence of elastic displacements, which are obtained by minimizing an energy functional based on the distance between the two shapes. The proposed method has been implemented in a finite elements setting and numerical examples in two and three dimensions are presented to illustrate its efficiency.

To cite this article: M. de Buhan, C. Dapogny, P. Frey, C. Nardoni, C. R. Acad. Sci. Paris, Ser. I 340 (2016).

Résumé

Dans cette note, nous nous intéressons au problème d'appariement de formes suivant : étant donné une forme de référence, représentée numériquement par un maillage de calcul, et une forme cible, connue seulement par l'intermédiaire de la fonction de distance signée à celle-ci, notre objectif consiste à déformer itérativement le maillage de la forme de référence en un maillage de la forme cible. Pour ce faire, nous nous appuyons sur des techniques d'optimisation de formes. Sous l'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'une énergie basée sur la distance entre les formes. La méthode a été implémentée en deux et trois dimensions d'espace et nous présentons des exemples numériques permettant d'apprécier son efficacité.

Pour citer cet article : M. de Buhan, C. Dapogny, P. Frey, C. Nardoni, C. R. Acad. Sci. Paris, Ser. I 340 (2016).

Email addresses: maya.de-buhan@parisdescartes.fr (Maya de Buhan), charles.dapogny@imag.fr (Charles Dapogny), frey@ann.jussieu.fr (Pascal Frey), chiara.nardoni@upmc.fr (Chiara Nardoni).

1 . This work received a financial support by Idex Sorbonne Universités under the French funds "Investissement d'Avenir", reference ANR-11-IDEX-0004-02.

1. Introduction

Shape morphing or matching arises in a wide variety of situations in areas from biomedical engineering to computer graphics and scientific computing. Beyond the specific stakes to each particular application, the general issue is to find one transformation from a given 'template' shape Ω_0 into a 'target' 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, reconstruction, or shape simplification). See for instance [15] and references therein for an overview of several related applications.

Understandably enough, a great deal of work has been devoted to shape matching, and we limit ourselves to mentioning a few approaches. In [5], the authors start by distributing sample points on the contour of both shapes, that will be matched according to their 'shape context'. They eventually infer a global transformation from this point-to-point correspondance. In the field of computational anatomy, a series of articles (see e.g. [4,8,11]) have suggested to describe a sought diffeomorphism between Ω_0 and Ω_T as the flow of a velocity field v, and to cast the search for v as an optimal control problem. The resulting mapping is used to study features of organs, detect anomalies, etc. 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 [13].

This note addresses the following problem: 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. The precise range of applications we have in mind will be described in a forthcoming, longer article.

To achieve our purpose, we rely on a method which has much in common with that of [2], 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.

2. Presentation of the method

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 are not necessarily close one from another. Our purpose is to map Ω_0 onto Ω_T, which we achieve numerically by deforming a mesh $\mathcal{T}_0$ of Ω_0 into one for Ω_T. For this purpose, we rely on the shape optimization setting.

2.1. *Shape matching as a shape optimization problem*

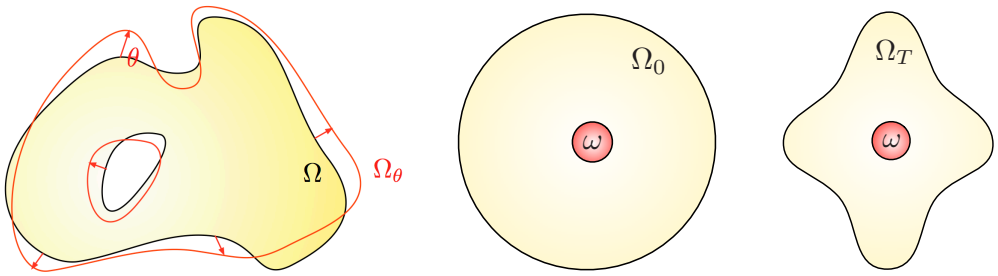
The discrepancy between a reference shape Ω and a target shape Ω_T is measured by the following functional $J(\Omega)$ of the domain:

$$J(\Omega) = \int_\Omega d_{\Omega_T}(x)dx, \tag{1}$$

which involves the Euclidean *signed distance function* d_{Ω_T} to Ω_T, defined as:

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

In the above formula, $d(\cdot, \partial\Omega_T)$ denotes the usual Euclidean distance function to $\partial\Omega_T$.



(a) (b)

Figure 1. (a) Variation Ω_θ of a shape Ω according to Hadamard's method; (b)Target and template shapes sharing a common fixed subset 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. Note that the functional $J(\Omega)$ has a unique, global minimizer $\Omega = \Omega_T$, and no extra local minimum point provided Ω_T is connected. It is then expected that an iterative (e.g. gradient-based) algorithm for minimizing $J(\Omega)$, starting from Ω_0, will lead to an interesting way to transform Ω_0 into Ω_T.

2.2. *Shape derivative of the functional $J(\Omega)$*

Several notions of differentiation with respect to the domain are available in the literature. One which is very-well tailored for our purpose is Hadamard's boundary variation method (see e.g. [1,12,14]), whereby variations of a bounded, Lipschitz domain $\Omega \subset \mathbb{R}^d$ are considered under the form (see Figure 1, left):

$$\Omega_\theta = (I + \theta)(\Omega), \;\; \theta \in W^{1,\infty}(\mathbb{R}^d, \mathbb{R}^d).$$

Accordingly, 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:

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

By a classical calculation, the shape derivative of the function $J(\Omega)$ defined in (1) reads:

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

This paves the way for an iterative algorithm, producing 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 step, Ω_k is updated according to

$$\Omega_{k+1} = (I + \theta_k)(\Omega_k), \;\; \text{where } \theta_k \text{ is (an extension to } \Omega_k \text{ of) } \; - d_{\Omega_T} n_{\Omega_k}, \tag{2}$$

and n_{Ω_k} stands for the unit normal vector to $\partial\Omega_k$, pointing outward Ω_k. The mesh $\mathcal{T}_k$ is updated as:

$$\forall x \text{ vertex of } \mathcal{T}_k, \; x \mapsto x + \theta_k(x). \tag{3}$$

2.3. *Parametrization by elastic displacements*

The formal procedure summarized in (2) boils down to deforming a shape Ω in the negative direction of the $L^2(\partial\Omega)^d$ gradient of the differential $\theta \mapsto J'(\Omega)(\theta)$. Unfortunately, this reveals unsuited when it comes to deforming a mesh $\mathcal{T}$ of Ω by moving its vertices. Indeed, the vector field $\theta = -d_{\Omega_T} n$ featured in (2) 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 $\mathcal{T}$. Moreover, 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 of $\mathcal{T}$ via (3) impossible to achieve without invalidating the mesh.

These difficulties can be alleviated by using the gradient of $\theta \mapsto J'(\Omega)(\theta)$ associated to another inner product. This *velocity extension - regularization* issue is quite classical in shape optimization (see [10] and references therein), and can be thought of as an efficient preconditioning of the naive procedure (2).

In the present context, imagine that all the considered shapes Ω are filled with a linear elastic material. Also, assume that any such shape Ω contains a given subset $\omega \Subset \Omega$ on which it is clamped; from the numerical point of view, the choice of such a subset corresponds to a global alignment of shapes (cf. Figure 1). We now obtain a descent direction for $J(\Omega)$ as the unique solution u_Ω of the linearized elasticity system (hereafter written in variational form)

$$\forall v \in H^1_\omega(\Omega)^d, \quad \int_\Omega \sigma(u_\Omega) : \varepsilon(v)\, dx = -J'(\Omega)(v) = -\int_{\partial\Omega} d_{\Omega_T} v \cdot n\, ds, \tag{4}$$

where $\varepsilon(u) = \frac{1}{2}(\nabla u + \nabla u^T)$ is the linearized strain tensor and σ is the associated stress tensor via the Hooke's law with Lamé coefficients λ, μ:

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

This vector field u_Ω is naturally a descent direction for $J(\Omega)$, and its advantages over the 'natural' deformation field θ defined in (2) are twofold:

(i) 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 [7]),

(ii) 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 (3); see e.g. [3] for an example of use of elastic displacements in the context of mesh displacement.

3. 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), 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 gradient descent algorithm in order to get a sequence of pairs $(\Omega_k, \mathcal{T}_k)$ of domains and their corresponding meshes with decreasing values of $J(\Omega_k)$.

Remark 1 *(i) 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.).*

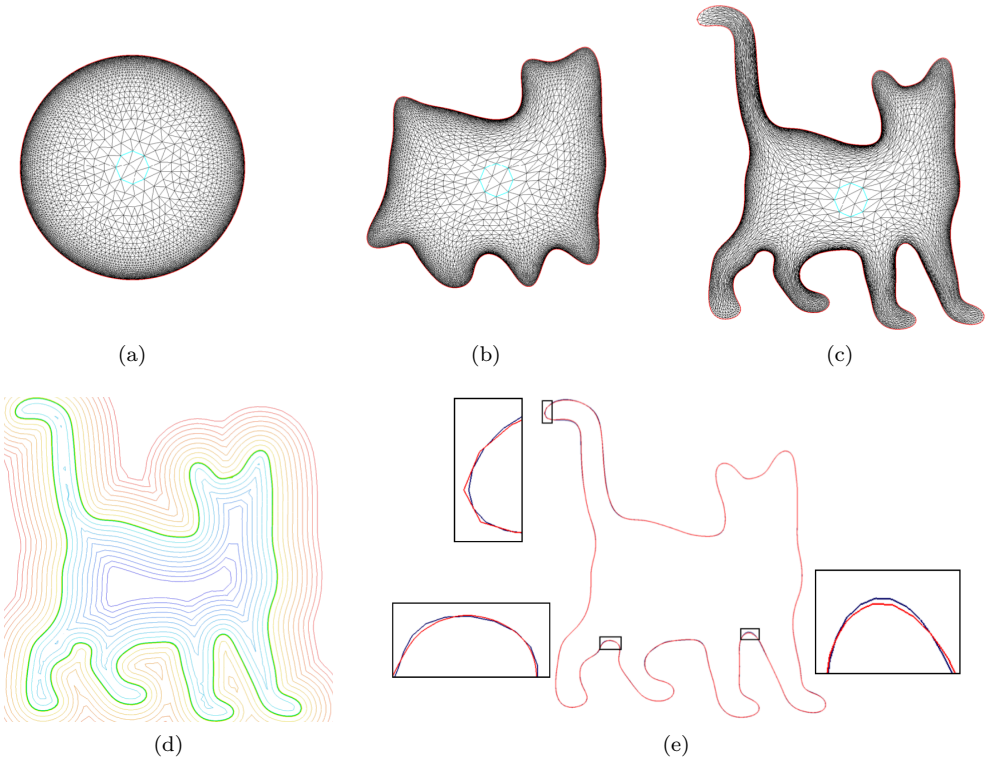
(ii) The global mapping from Ω_0 to Ω_T is easily recovered by the composition of the different displacements between each iteration.

(iii) 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.

3.1. *Numerical examples*

In the proposed examples, the calculation of the signed distance function to Ω_T is performed using the algorithm described in [9]. At first, Figure 2 depicts a $2d$ test case. 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 2 100 iterations. The L^2 norm of the distance d_{Ω_T} calculated on the boundary of the resulting shape Ω_{2100} equals $5.73e^{-4}$ (much smaller than the minimal mesh size), revealing an excellent matching of Ω_T.

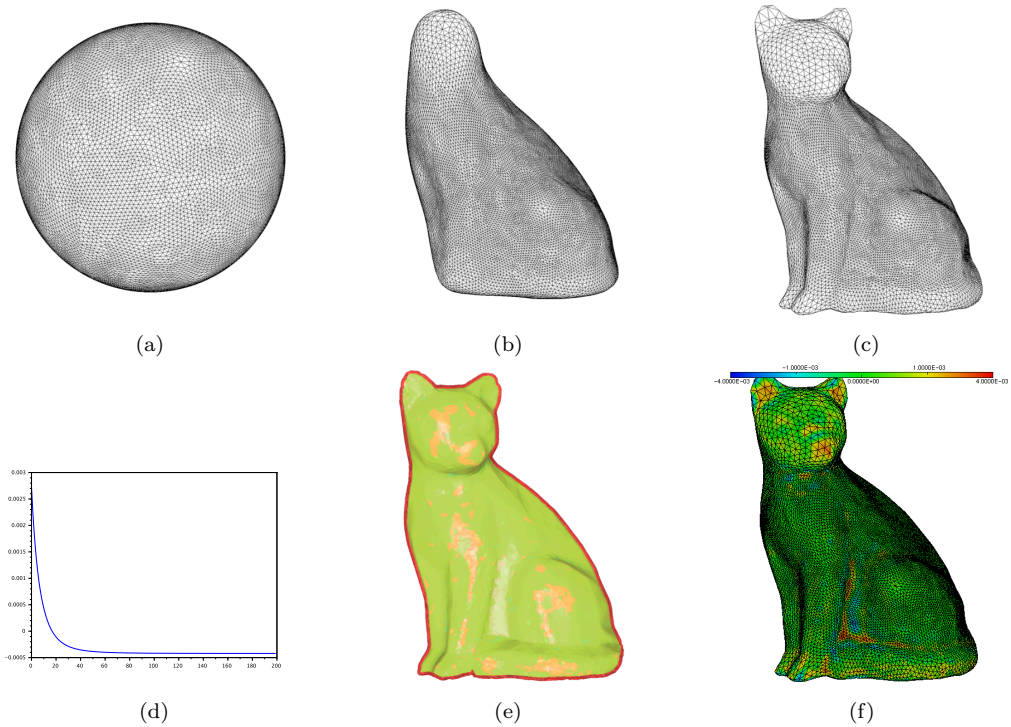
Next, we consider a $3d$ example; see Figure 3. Both the target and the template meshes are embedded in a unit computational box $D = [0,1]^3$. The shapes Ω_0 and Ω_T are aligned by choosing a small ball ω



(a) (b) (c)

(d) (e)

Figure 2. An example in 2D: (a) Template shape Ω_0. (b) Deformed shape Ω_k for $k = 90$. (c) Deformed shape Ω_k for $k = 2100$. (d) Isovalues of the signed distance function to the target shape Ω_T defined on the fixed mesh $\mathcal{T}_D$. (e) Discrepancy between Ω_T and Ω_{2100}.



(a) (b) (c)

(d) (e) (f)

Figure 3. An example in 3D: (a) Template shape Ω_0. (b) Deformed shape Ω_k for $k = 300$. (c) Deformed shape Ω_k for $k = 2250$. (d) Objective functional $J(\Omega_k)$ versus number of iterations k. (e) Target shape Ω_T as the zero level set of the signed distance function. (f) Discrepancy between Ω_T and Ω_{2250}.

in $\Omega_0 \cap \Omega_T$ as for the subset ω. The template mesh $\mathcal{T}_0$ has about 9 000 triangles, and 2 250 iterations of the gradient descent algorithm have been performed to achieve convergence, running in a few minutes on a standard laptop computer. The L^2 norm of the distance d_{Ω_T} calculated on the boundary of the final shape Ω_{2250} is $5.04e^{-4}$ (again, much smaller than the minimal mesh size).

References

[1] G. Allaire, *Conception optimale de structures*, Mathématiques & Applications, **58**, Springer Verlag, Heidelberg (2006).

[2] R. Bajcsy and S. Kovacic *Multiresolution elastic matching*, Computer Vision, Graphics and Image Processing, 46, (1989) pp. 1–21.

[3] T. J. Baker, *Mesh Movement and Metamorphosis*, Eng. Comput. 18, 1, (2002), pp. 188–198.

[4] M.F. Beg, M.I. Miller, A. Trouvé and L. Younes, *Computing large deformation metric mappings via geodesic flows of diffeomorphisms*, Int. J. Comput. Vis., 61, (2005), pp. 139–157.

[5] S. Belongie, J. Malik, and J. Puzicha, *Shape matching and object recognition using shape contexts*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (4), (2002), pp. 509–522..

[6] J. Céa, *Conception optimale ou identification de formes, calcul rapide de la dérivée directionnelle de la fonction coût*, Math. Model. Num. 20, 3 (1986), pp. 371–420.

[7] P.G. Ciarlet, *Mathematical Elasticity, vol I: Three Dimensional Elasticity*, North Holland Publishing Company (1988).

[8] P. Dupuis, U. Grenander, and M.I. Miller, *Variational problems on flows of diffeomorphisms for image matching*, Q. Appl. Math., 56, (1998), pp. 587–600.

[9] C. Dapogny and P. Frey, *Computation of the signed distance function to a discrete contour on adapted triangulation*, Calcolo, Volume 49, Issue 3, pp. 193-219 (2012).

[10] F. de Gournay, *Velocity extension for the level-set method and multiple eigenvalues in shape optimization.* SIAM J. on Control and Optim., 45, no. 1, 343–367 (2006).

[11] U. Grenander and M.I. Miller, *Computational anatomy: an emerging discipline*, Current and future challenges in the applications of mathematics (Providence, RI, 1997), Quart. Appl. Math. 56, no. 4, (1998), pp. 617–694.

[12] A. Henrot and M. Pierre, *Variation et optimisation de formes, une analyse géométrique*, Mathématiques et Applications 48, Springer, Heidelberg, (2005).

[13] B. Lévy, *A numerical algorithm for L_2 semi-discrete optimal transport in 3D*, ESAIM: Math. Model. Numer. Anal., 49, 6, (2015), pp. 1693–1715.

[14] F. Murat and J. Simon, *Sur le contrôle par un domaine géométrique*, Technical Report RR-76015, Laboratoire d'Analyse Numérique (1976).

[15] R.C. Veltkamp, *Shape matching: Similarity measures and algorithms*, In Shape Modeling and Applications, SMI 2001, IEEE, (2001), pp. 188–197.