Computational patient avatars for surgery planning

David González[1], Elías Cueto[1*], and Francisco Chinesta[2]

[1]Aragon Institute of Engineering Research. Universidad de Zaragoza, Zaragoza, Spain

[2]Ecole Centrale de Nantes, Nantes, France.

[*]To whom correspondence should be addressed; E-mail: `ecueto@unizar.es`.

Abstract

In this paper a new method is described for the generation of computational patient avatars for surgery planning. By "patient avatar" a computational, patient-specific, model of the patient is meant, that should be able to provide the surgeon with an adequate response under real-time restrictions, possibly including haptic response. The method is based on the use of *computational vademecums* [F. Chinesta et al., PGD-based computational vademecum for efficient design, optimization and control. Archives of Computational Methods in Engineering, 20(1), 31-59, 2013.] that are properly interpolated so as to generate a patient-specific model. It is highlighted how the interpolation of *shapes* needs for a specialized technique, since a direct interpolation of biological shapes would produce, in general, non-physiological shapes. To this end a manifold learning technique is employed, that allows for a proper interpolation that provides very accurate results in describing patient-specific organ geometries. These interpolated vademecums thus give rise to very accurate patient avatars able to run at kHz feedback rates, enabling not only visual, but also haptic response to the surgeon.

Keywords: Computational surgery, surgery planning, patient specific, real-time simulation.

1 Introduction

Systems patientomics has been recently defined [18] as a systematic approach to the generation of virtual *in silico* patients, able to provide, by means of an integration of patient- and population-specific data, essential predictions to clinicians in general and surgeons in particular. It is envisaged that computational sciences will be able, in a not-so-far future, to provide us with predictive informations about decisions to make, and that these decisions will be taken upon personalized computational models, the so-called patient avatars. These avatars would comprise information across multiple scales, from purely macroscopic mechanical properties to gene regulatory systems.

In this work an advancement in the development of patient avatars with an eye towards personalized surgery planning simulators is made. The goal is the development of patient-specific surgery planning systems able to provide the surgeon with valuable information on the patient's anatomy and its mechanical response so as to be able to forecast the difficulties that most likely will be faced during the real surgery procedure. At this level only purely mechanical effects will be taken into account but, as will be discussed later, information at all levels (including gene regulatory systems) could be handled under the same rationale in a multiscale framework, see [4].

The challenge of a patient-specific simulator

The development of a surgery simulator faces several difficulties, see [15]. To this end, several approaches can be found in the literature [22] [16] [12] [6] [17] [13], although they are in general limited to very simplified constitutive modeling of soft tissues. Only recently, the so-called Total Lagrangian Explicit Dynamics approach has been able to develop explicit finite element techniques including hyperelasticity with haptic feedback [33] [23] [14].

Even though a realistic-enough surgery simulator has not been completely achieved, the consideration of patient-specific information still complicates the challenge. First of all, personalized geometry should be obtained from any type of medical imaging system (computerized tomography, ultrasounds, etc.) and conveniently segmented so as to provide an accurate description of the patient's anatomy. This is not, however, the objective of this work. The interested reader could consult [32] for an excellent review on the topic.

Once the patient's anatomy has been segmented, a computational model able to provide the mechanical response of the different organs with a relevant role during the surgical procedure must be developed. The approach here considered to this problem is conceptually simple: to interpolate the patient-specific anatomy from previously computed *computational vademecums* [9] for the organ(s) of interest. In engineering practice, the use of vademecums (or compilations of known solutions to common problems) has a strong tradition, with the Bernoulli vademecum as the most typical example one can cite [5]. Recently, the concept has been updated by the authors, by adding a computational character. Thus, off-line computed numerical simulations of parametric problems are exploited in real time, even with handheld, deployed platforms such as smartphones or tablets [1].

In general, the computational solution of parametric problems needs for a campaign of (numerical) experiments and a sort of interpolation between the solutions so as to obtain the desired *response surface* of the parametric model. It is well-known, however, that it is not possible to "mesh" the entire parametric domain and just to solve the resulting high-dimensional problem. Since the number of degrees of freedom grows exponentially with the number of dimensions of the phase space, the size of the problem soon becomes intractable, giving rise to the so-called *curse of dimensionality* [20].

Recently a technique to overcome these problems has been proposed. Considered as a generalization of Proper Orthogonal Decomposition (POD, also known as Principal Component Analysis, or Karhunen-Loeve transform), the Proper Generalized Decomposition (PGD) [10] [8] [11] approximates the solution in the form of a finite series of separated functions, thus leading in practice to the solution of a series of (non-linear) low-dimensional problems rather than a high-dimensional one. PGD techniques have been employed in the past by the authors to produce computational vademecums to develop haptic surgery simulators, see for instance, [25] [26].

The procedure here proposed is thus conceptually simple. From a database of pre-computed vademecums that provide the response of the organ of interest for any value (within a prescribed interval) of the parameters of the model, the just segmented anatomy of the patient is used to obtain a new interpolated vademecum. Parameters could be material coefficients, thus taking into account patient variability of the properties, position of the loads provoked by the interaction with the surgeon scalpel, or, in general, any other.

As mentioned before, the obtention of a single vademecum is not an easy task, nor their interpolation. It has been noticed that the anatomy can not be straightforwardly interpolated. Rather, more sophisticated manifold learning techniques must be employed, since the interpolation of two physiological anatomies does not correspond, in general, to a physiological one [30].

The structure of the paper is as follows. In Section 2.1 a brief overview of the concept of computational vademecum, which is not new nor the objective of this paper, is made. It is included here for completeness, although the interested reader can consult [25] [26] [19] for more details. Then, in Section 2.2 manifold learning techniques are introduced as a means of interpolating anatomies in a rigorous way. In Section 3 some results that prove the validity of the proposed method are shown. First, a set of *in silico*-generated anatomies are analyzed, and finally a set of clinical data is employed. The paper is closed in Section 4 with a discussion on the accuracy of the results.

2 Materials and methods

2.1 A brief overview of computational vademecums for haptic surgery simulation

In this section we reproduce, for completeness, results from some of the authors' previous works on the topic. Essentially, to develop a surgery simulator, we look for the solution of the mechanical problem under any possible location of the force of contact between surgery tool and organ, say $\boldsymbol{s}$. Consider, for simplicity, the static equilibrium equations of a general solid under small strain assumptions:

$$\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{b} = \mathbf{0} \text{ in } \Omega, \tag{1}$$

where $\boldsymbol{b}$ represents the volumetric forces applied to the body, subjected to the following boundary conditions

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \text{ on } \Gamma_u \tag{2}$$

$$\boldsymbol{\sigma n} = \bar{\boldsymbol{t}} \text{ on } \Gamma_t \tag{3}$$

The standard weak form of the problem is obtained after multiplying both sides of Eq. (1) by an admissible variation of the displacement, $\boldsymbol{u}^*$, and integrating over the domain Ω. The basic ingredient of the PGD to the problem is to consider the load $\bar{\boldsymbol{t}}$ as a parameter of the formulation, thus enabling to obtain a parametric response surface to the problem. The load is assumed to be applied at any point $\boldsymbol{s}$ of the surface and, for the sake of simplicity in the exposition, unitary and vertical. Then, $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{s}) \in \Omega \times \bar{\Gamma}$, where $\bar{\Gamma} \subseteq \Gamma_t$ represents the portion of the boundary of the organ where the load can be applied (region accesible to the surgeon).

To account for the high dimensionality of the problem, an alternative (doubly) weak form of problem (1)-(3) consists in finding the displacement $\boldsymbol{u} \in \mathcal{H}^1(\Omega) \times L_2(\bar{\Gamma})$ such that for all $\boldsymbol{u}^* \in \mathcal{H}_0^1(\Omega) \times L_2(\bar{\Gamma})$ (see [25]):

$$\int_{\bar{\Gamma}} \int_{\Omega} (\boldsymbol{\nabla}_s \boldsymbol{u}^*)^T \boldsymbol{\sigma} d\Omega d\bar{\Gamma} = \int_{\bar{\Gamma}} \int_{\Gamma_{t2}} (\boldsymbol{u}^*)^T \boldsymbol{t} d\Gamma d\bar{\Gamma} \tag{4}$$

where $\boldsymbol{\nabla}_s \boldsymbol{u}$ represents the symmetric part of the gradient of displacements, $\Gamma = \Gamma_u \cup \Gamma_t$ represents the boundary of the solid, divided into essential and natural regions, and where $\Gamma_t = \Gamma_{t1} \cup \Gamma_{t2}$, i.e., regions of homogeneous and non-homogeneous, respectively, natural boundary conditions.

The Dirac-delta term should be regularized for computation purposes and approximated by:

$$t_j \approx \sum_{i=1}^{m} f_j^i(\boldsymbol{x}) g_j^i(\boldsymbol{s}) \tag{5}$$

by performing a singular value decomposition of the load, for instance. Here, j refers to the j-th component of the vector at hand, and m the truncation index —the series would be in principle infinite— whose value should be determined with the help of a suitable error indicator, see for instance [3].

The main ingredient of the PGD approach to the construction of a computational vademecum is the establishment, in an iterative way, of an approximation to the solution in the form of a finite sum of separable functions [10]. To briefly describe the method, we assume that, at iteration m of this procedure the solution has converged, and takes the form

$$u_j^m(\boldsymbol{x}, \boldsymbol{s}) = \sum_{k=1}^{m} X_j^k(\boldsymbol{x}) \cdot Y_j^k(\boldsymbol{s}), \tag{6}$$

where the term u_j refers to the j-th component of the displacement vector, $j = 1, 2, 3$. Here, vectorial functions $\boldsymbol{X}^k$ and $\boldsymbol{Y}^k$ represent a set of a priori unknown approximating functions that will be computed by the PGD algorithm with no user intervention. In general, and perhaps by the strong tradition of POD methods, they are referred to as *modes* also in the PGD terminology.

The subsequent term of this approximation, the $(m+1)$-th one, will be obtained by means of an enrichment with space and load position functions, say

$$u_j^{m+1}(\boldsymbol{x},\boldsymbol{s}) = u_j^m(\boldsymbol{x},\boldsymbol{s}) + R_j(\boldsymbol{x}) \cdot S_j(\boldsymbol{s}), \tag{7}$$

where $\boldsymbol{R}(\boldsymbol{x}) = \boldsymbol{X}^{m+1}(\boldsymbol{x})$ and $\boldsymbol{S}(\boldsymbol{s}) = \boldsymbol{Y}^{m+1}(\boldsymbol{s})$ are the sought functions that improve the approximation, here renamed for brevity. The admissible variation of the displacement is now

$$u_j^*(\boldsymbol{x},\boldsymbol{s}) = R_j^*(\boldsymbol{x}) \cdot S_j(\boldsymbol{s}) + R_j(\boldsymbol{x}) \cdot S_j^*(\boldsymbol{s}). \tag{8}$$

To find the new pair of functions R_j and S_j, our experience indicates that a fixed-point alternating directions algorithm, in which functions R_j and S_j are sought iteratively, gives excellent results despite its simplicity. Newton-Raphson or modified Newton methods could be equally employed.

2.1.1 Computation of $S(\boldsymbol{s})$ assuming $R(\boldsymbol{x})$ is known

If $\boldsymbol{R}(\boldsymbol{x})$ is assumed to be known, the resulting admissible variation will include terms related to $\boldsymbol{S}(\boldsymbol{s})$ only:

$$u_j^*(\boldsymbol{x},\boldsymbol{s}) = R_j(\boldsymbol{x}) \cdot S_j^*(\boldsymbol{s}), \tag{9}$$

or, in vector notation, $\boldsymbol{u}^*(\boldsymbol{x},\boldsymbol{s}) = \boldsymbol{R} \circ \boldsymbol{S}^*$, where the symbol "$\circ$" stands for the so-called entry-wise, Hadamard or Schur multiplication for vectors. Injecting it back into the weak form of the problem, Eq. (4), provides

$$\int_{\bar{\Gamma}} \int_{\Omega} \boldsymbol{\nabla}_s(\boldsymbol{R} \circ \boldsymbol{S}^*) : \mathbf{C} : \boldsymbol{\nabla}_s \left(\sum_{k=1}^{m} \boldsymbol{X}^k \circ \boldsymbol{Y}^k + \boldsymbol{R} \circ \boldsymbol{S} \right) d\Omega d\bar{\Gamma} = \int_{\bar{\Gamma}} \int_{\Gamma_{t2}} (\boldsymbol{R} \circ \boldsymbol{S}^*) \cdot \left(\sum_{k=1}^{m} \boldsymbol{f}^k \circ \boldsymbol{g}^k \right) d\Gamma d\bar{\Gamma}, \tag{10}$$

or, equivalently (functional dependencies have been omitted for clarity when they are obvious)

$$\int_{\bar{\Gamma}}\int_{\Omega}\boldsymbol{\nabla}_s(\boldsymbol{R}\circ\boldsymbol{S}^*):\mathbf{C}:\boldsymbol{\nabla}_s(\boldsymbol{R}\circ\boldsymbol{S})d\Omega d\bar{\Gamma}$$
$$=\int_{\bar{\Gamma}}\int_{\Gamma_{t2}}(\boldsymbol{R}\circ\boldsymbol{S}^*)\cdot\left(\sum_{k=1}^{m}\boldsymbol{f}^k\circ\boldsymbol{g}^k\right)d\Gamma d\bar{\Gamma} \tag{11}$$
$$-\int_{\bar{\Gamma}}\int_{\Omega}\boldsymbol{\nabla}_s(\boldsymbol{R}\circ\boldsymbol{S}^*)\cdot\mathcal{R}^n d\Omega d\bar{\Gamma},$$

where $\mathcal{R}^n$ stands for the residual:

$$\mathcal{R}^n=\mathbf{C}:\boldsymbol{\nabla}_s\boldsymbol{u}^m, \tag{12}$$

and $\mathbf{C}$ represents, as usual, the fourth-order constitutive tensor of the particular model employed for soft tissues.

Since all the terms depending on $\boldsymbol{x}$ are known at this stage, all integrals over Ω and Γ_{t2} can be computed straightforwardly so as to obtain an equation to compute $\boldsymbol{S}(\boldsymbol{s})$.

2.1.2 Computation of $R(\boldsymbol{x})$ assuming $S(\boldsymbol{s})$ is known

As the fixed-point algorithm proceeds, an equivalent procedure allows us to obtain $\boldsymbol{R}(\boldsymbol{x})$ by noting that now $\boldsymbol{S}(\boldsymbol{s})$ is assumed to be known:

$$u_j^*(\boldsymbol{x},\boldsymbol{s})=R_j^*(\boldsymbol{x})\cdot S_j(\boldsymbol{s}). \tag{13}$$

Again, by substitution into the weak form of the problem, Eq. (4), we obtain

$$\int_{\bar{\Gamma}}\int_{\Omega}\boldsymbol{\nabla}_s(\boldsymbol{R}^*\circ\boldsymbol{S}):\mathbf{C}:\boldsymbol{\nabla}_s\left(\sum_{k=1}^{m}\boldsymbol{X}^k\circ\boldsymbol{Y}^k+\boldsymbol{R}\circ\boldsymbol{S}\right)d\Omega d\bar{\Gamma}=$$
$$\int_{\bar{\Gamma}}\int_{\Gamma_{t2}}(\boldsymbol{R}^*\circ\boldsymbol{S})\cdot\left(\sum_{k=1}^{m}\boldsymbol{f}^k\circ\boldsymbol{g}^k\right)d\Gamma d\bar{\Gamma}. \tag{14}$$

Conversely, all the terms depending on s can now be integrated over $\bar{\Gamma}$, thus arriving at a system of equations to determine $\boldsymbol{R}(\boldsymbol{x})$.

Based on this procedure, a surgery simulator that provides very accurate response and that includes haptic feedback, has been constructed, see Fig. 1. Contrarily to what is commonly believed, the number of *modes* (i.e., the number of functions, m), needed for the approximation of the response is not that big. In practice, we have been employing in the order of tens of functional pairs [25], although an error estimator is being currently developed that could relate the number of functions to the error in the force feedback provided to the user.

[Figure 1 about here.]

2.2 Manifold learning techniques for the interpolation of anatomies

As can be noticed from the introduction before, what PGD technique is intended for is the solution of parametric problems, considered as problems in high dimensional spaces. Therefore, it seems natural to think of an extension in which "shape" itself is a parameter of the problem so as to be able to provide the user with solutions to any possible (physiological) anatomy. The inclusion of the patient anatomy as a parameter in the formulation is a delicate task, however, as will be readily noticed. *Shape* is not a parameter in classical terms. In other words, by linearly interpolating two ellipses, for instance, shapes very different to an ellipse can be obtained. This is a very well-known phenomenon that has already been pointed out in many branches of science and engineering [28] [29].

To efficiently parameterize *shapes*, following [28], it has been preferred to embed the segmented organ(s) geometry (a human liver in this case) on a background mesh and to compute in it the distance field to the boundary of the organ (in other words, a level set), see Fig. 2. This approach is also classical in shape and topology optimization [7] [2], where the concept of shape derivative and also topological derivative is introduced. The set of (Euclidean) nodal distances is thus stored in the form of a high dimensional vector, living in the example of Fig. 2 in a space of 49321 dimensions, that correspond to the distance values at the nodal positions of a grid of $43 \times 31 \times 37$ nodes. Therefore, each anatomy represents a point in this 49321-d space. Associated to each anatomy, an off-line pre-computed vademecum is considered that provides with its response with respect to a punctual load at any point of its accesible boundary, as introduced in the previous section.

[Figure 2 about here.]

To find the underlying geometrical structure of a particular set of organ anatomies and their associated, pre-computed vademecums, Locally Linear Embedding (LLE) techniques have been employed [30]. Essentially, LLE looks for a suitable, lower-dimensional space where to project the set of high dimensional vectors and still obtain meaningful results.

Denote by $\boldsymbol{X}_i$ the coordinates of each organ in the high dimensional space, that is, its associated 49321-dimensional vector containing the distance to the organ boundary. LLE looks for a suitable interpolation of each point i in terms of a number of neighbors (that must be chosen by the user, always greater than the expected dimension of the low dimensional space where we try to project). These weights are found by minimizing the functional

$$\varepsilon(\boldsymbol{W}) = \sum_i |\boldsymbol{X}_i - \sum_j W_{ij} \boldsymbol{X}_j|^2, \tag{15}$$

where $W_{ij} = 0$ if i and j are not neighbors. The basic assumption of LLE is that, in the low dimensional space, these weights still interpolate well the new, embedded coordinates $\boldsymbol{Y}_i$. These new, low dimensional coordinates are thus found by minimizing a new functional,

$$\Phi(\boldsymbol{Y}) = \sum_i |\boldsymbol{Y}_i - \sum_j W_{ij} \boldsymbol{Y}_j|^2, \tag{16}$$

that leads to a problem in which $\Phi(\boldsymbol{Y}) = \sum_{ij} M_{ij}(\boldsymbol{Y}_i \cdot \boldsymbol{Y}_j)$ with $M_{ij} = \delta_{ij} - W_{ij} - W_{ji} - \sum_k W_{ki}W_{kj}$. Precisely, see [30], the lowest eigenvalues of this matrix M_{ij} serve as an indirect measure of the dimensionality of the embedding space (except from the first eigenvector, always unitary, with null associated eigenvalue, that is discarded).

By embedding the geometry of a particular organ in the manifold generated by other organs, also obtained from any suitable medical imaging technique and after segmentation, it will be shown that it is possible to efficiently characterize the underlying geometry and, notably, to interpolate it among their neighbors.

3 Results

Before giving details on the performance of the proposed technique, we revisit the performance of PGD as a menas to obtain suitable vademecums for the organ(s) of interest. Then, the resulting performance of the proposed technique for interpolating vademecums is addressed.

3.1 On the convergence of the PGD technique

It is firstly important to clarify the error introduced by the truncation of the PGD series at a number m of functional pairs, see Section 2.1. Although it is a classical result from some of our previous works, it is important to evaluate all sources of error.

In this example we consider one particular liver geometry, the same employed in some of our previous works, see [25], for instance. The mesh is composed by 2853 nodes and 10519 tetrahedra. For this simple example, a Kirchhoff-Saint Venant constitute law with E $= 0.17$ MPa and $\nu = 0.48$ is considered. More sophisticated constitutive laws are equally possible, see [15] and references therein. To evaluate the convergence properties of the method, we consider a loading region $\bar{\Gamma}$ composed by 66 nodes. The explicit algorithm introduced in this same reference [25] is also considered. In Fig. 3 a plot of the convergence in the error with respect to the full finite element solution of the problem (for one particular position of the load, randomly chosen) is represented. As can be noticed, some eleven modes suffice to obtain errors below 10% in $L2$-norm, a value that seems judicious, given the inherent uncertainties of this type of application. Nevertheless, there is no limitation in terms of real-time performance and storage of a much higher number of modes, if needed.

[Figure 3 about here.]

Once the tolerance has been fixed for a particular basis set of vademecums, the patient-specific one is then obtained by a suitable interpolation on the shape manifold, as explained before. This is analyzed next.

3.2 In-silico generated anatomies

It is not easy, in general, to have access to a big enough number of organ geometries. Although in Section 3.3 an example will be given, to show the capabilities of the proposed method it has been preferred to begin with a set of synthetic geometries. Thus, by beginning with a segmented geometry of a human liver, a set of 50 different livers has been generated by applying to its original finite element mesh a transformation

$$\boldsymbol{x}' = \boldsymbol{A}\boldsymbol{x} = \begin{pmatrix} \alpha_x & \beta_{xy} & \beta_{xz} \\ & \alpha_y & \beta_{yz} \\ & & \alpha_z \end{pmatrix} \boldsymbol{x}. \tag{17}$$

These α_i and β_{ij} parameters have been randomly chosen such that the volume gain (i.e., $\sum_i \alpha_i$) is always under 30% and that the distortion provoked by β_{ij} is always under 10%. The resulting livers are shown in Fig. 4.

[Figure 4 about here.]

As can be noticed from Eq. (17), every liver from the set is identical to the original one except a six-parameter transformation given by the symmetric matrix $\boldsymbol{A}$. Thus, it is expected that LLE method will be able to find this relationship. Indeed, by applying LLE, the set of eigenvalues clearly show that every point in the manifold (a liver in the form of its 49321-d level set nodal vector) depends on six values, the six eigenvalues clearly distant from the others (here, the already mentioned null eigenvalue is discarded, see [27]). The set of eigenvalues and the projection of the set of 50 liver nodal coordinates into a three-dimensional space by LLE are shown in Fig. 5(a) and (b), respectively. It is noteworthy to mention that upon projection onto a 3-d space, still a meaningful structure of the set is found, see Fig. 5(b), where the set fits perfectly on a plane that could even be projected onto a two-dimensional space almost without distortion. This fact is suggested by the two eigenvalues shown in Fig. 5(a), clearly distant from the others.

[Figure 5 about here.]

With an eye towards the interpolation of the computational vademecums generated on top of this geometric structure, a total of 45 livers were used to generated the manifold structure, while the remaining 5 livers of the set were utilized for testing the proposed interpolation method. Essentially, what is here proposed is to interpolate among vademecums $\boldsymbol{u}_i = \boldsymbol{u}_i(\boldsymbol{x}, \boldsymbol{s})$, with $i = 1, \ldots, 45$, by simply applying the LLE structure:

$$\boldsymbol{u}_j(\boldsymbol{x}, \boldsymbol{s}) \approx \sum_{i}^{nn} W_{ij} \boldsymbol{u}_i(\boldsymbol{x}, \boldsymbol{s}), \tag{18}$$

where W_{ij} represent the set of weights obtained by straightforward application of the LLE algorithm and nn the number of chosen neighbors (that is, as already mentioned, a user parameter). Note that the weights result from the interpolation of the geometry. We assume that the vademecum (i.e., the displacement field for each possible load position) will equally be well interpolated by these weights. Results indicate that this assumption works reasonably well.

By doing this, one of the vademecums was interpolated from among their neighbors, as dictated by the LLE algorithm, and the predicted results were compared to the reference ones, obtained by standard application of the method presented in Section 2.1. Results from the comparison between the reference anatomy and the just interpolated one are shown in Fig. 6(a). As can be noticed, very good agreement between both vademecums is obtained, with an error in the predicted volume of the liver under 5% and an error in the displacement field, measured in the L_2-norm, of 9.03%, see Fig. 6(b).

[Figure 6 about here.]

Another aspect of utmost importance in the application of LEE techniques is the expected rate of convergence of the method with the number of samples taken from the manifold. As intuition dictates, the more we sample the manifold, the more accurate the results should be. To verify this intuitive assumption, a set of up to 500 samples was generated. Fig. 7 plots the obtained convergence rate. As expected, the error decreases monotonically with the number of samples considered.

[Figure 7 about here.]

It is important to note that the number of available data could (and should) be increasing as new anatomies are being obtained by medical imaging techniques and progressively incorporated to the system database. Currently, there exist efficient incremental LLE algorithms able to improve the description of the manifold without the need for a full re-computing of the whole LLE weights, see Eq. (15)[31]. Ideally, *big data* streams coming continuously from CT scans can be incorporated into the high dimensional descriptions in the proposed data base, thus allowing for a finer description of the data manifold, especially in regions of intricate geometry.

3.3 In-vivo geometries

But the real challenge of the method here proposed is to work with segmented patient geometries. These are always harder to find. In this case, a set provided by the french research institute IRCAD, whose collaboration is gratefully acknowledged, was employed. The set was composed, see Fig. 8, by 20 segmented geometries, in `*.obj` format, corresponding to 10 men and 10 women, with different degrees of hepatic cancer. The tumor geometry has not been segmented, however, and for the ease of exposition has been treated as if it were healthy liver tissue. Consideration of a second distance field to properly take into account tumoral tissue, or even blood vessels, etc., could in principle be equally possible, although it has not been considered in this work.

[Figure 8 about here.]

3.3.1 Validation of the technique

The proposed technique has been validated, in the first instance, by evaluating the error in the results that the description of the anatomy by means of level sets implies. Of course, this error must be mesh-dependent, so four different grid sizes h were employed for a single liver anatomy. One of the liver anatomies, namely, that of liver number 2 in our nomenclature, was interpolated by employing a base composed by the original 20 livers. Therefore, liver number 2 was a member of the base. As expected, the LLE algorithm predicted that the weights $W_{ij} = 0 \ \forall i, j$ except from $W_{22} = 1.0$. In other words, LLE algorithms are able to detect the presence of the target anatomy in the basis and to interpolate it without adding any noise to the result. All the error must come indeed from the passage through a level set description of the anatomy and its associated vademecum.

The resulting anatomy is presented in Fig. 9. In this case, the error in the displacement field provided by the associated vademecum in L_2-norm is of 9% for a grid size of $h = 1$ mm. A convergence study with grid size is presented in this same Fig. 9.

[Figure 9 about here.]

3.3.2 Interpolating one target anatomy and its related vademecum

The true objective of the conducted research was to develop a method able to interpolate an anatomy and its associated vademecum by using a database composed by already computed vademecums on reference anatomies. To this end, one of the twenty livers (namely, liver number 13 in Fig. 8) was removed from the base and chosen as a target anatomy. With the aid of the remaining 19 livers and their respective vademecums, a model was created for liver number 13 including its interpolated anatomy and the associated vademecum.

By applying LLE algorithms, the anatomy of the liver was interpolated, giving the appearance shown in Fig. 10(a). In Fig. 10(b) the error thus provoked, measured as the difference between the interpolated and target level sets, is shown. Note that the difference between distance fields remains bounded, always below some 4 mm, within the region occupied by the liver.

[Figure 10 about here.]

By interpolating the vademecums using $d = 4$ dimensions and assuming $k = 10$ neighbors, the L_2-norm error in the displacement field for different load positions is below 19%. These values have been found by trial and error (with the obvious limitation that the number of neighbors should be greater than the number of dimensions in which the manifold lives), so as to provide the best possible results. Even if this error value is high, if we think in terms of usual engineering practice, it is not so if we consider all the uncertainties related to the correct constitutive modeling of soft living tissues. Furthermore, it should be highlighted that these results can be considered as a upper bound of the errors provided by the proposed technique, since they have been obtained for a dataset of only twenty livers, ten coming from male patients and ten from female, and with very different degrees of anatomy deformation provoked by different cancer types. One of the main advantages of the proposed techniques is precisely that it can be continuously improved by adding new organ segmentations to the data set. In this way, a better characterization of the underlying geometrical structure of the problem (i.e., the shape of the manifold) can be achieved without the need for re-computing the whole set of interpolating weights.

4 Discussion

In this work a technique has been developed that is able to compute computational, patient-specific, avatars for surgery planning, including haptic response. The just developed technique is based on the use of computational vademecums, i.e., a technique developed by the authors that can be seen as a sort of (entirely computational) response surface method that provides the user with the mechanical response of the organ(s) of interest at feedback rates on the order of 1 kHz, thus amenable to be employed in surgery planning environments with haptic response.

The systems is thus equipped with a database of (off-line) pre-computed vademecums for geometries of reference that serve as a basis for a proper interpolation of the target, patient-specific anatomy. The method is thus able to provide a patient-specific vademecum in a very short time, ranging from minutes to few hours. The main difficulty of such an interpolation is the fact that geometry can not be interpolated straightforwardly. Instead, a manifold learning technique should be employed. In this work, Locally Linear Embedding (LLE) techniques have been employed. This technique allows for a progressive enrichment of the database without the need of re-computation of the associated weights.

This is precisely one of the future enhancements of the technique that constitutes nowadays our main line of research. The work presented herein has been accomplished with a very restricted data set composed by only twenty livers, which provides limited accuracy in the interpolated vademecums. However, even in such a restrictive scenario, the presented technique has shown to properly interpolate patient anatomies and, perhaps more importantly, to provide with very reasonable results in very limited time windows. Results reported herein were obtained by employing a Mac Pro computer equipped with 6 Intel E5 cores and 16 Gb of RAM and by employing some of the Matlab parallelizing capabilities. No special supercomputing capabilities were needed, although for some very fine grids some high performance computing could be needed.

Computational costs of the proposed technique can be established clearly in two parts. Firstly, the obtain of the vademecums, which is by far the most costly part. In general, as reported in some of our previous works, one single vademecum could take up to some three-four days running in a standard Mac Pro computer, with no parallelization in the code. No supercomputing facilities were employed in this work.

Once obtained, vademecums should be interpolated through LLE techniques. This procedure runs considerably faster. For instance, in the example with 500 liver geometries only 1.4 seconds were necessary to run the LLE algorithm. This can be considered as a realistic upper bound of the consumed time.

The just developed technique thus opens the possibility for a very realistic patient-specific surgery planning that could eventually help the surgeon to perform the surgery virtually in advance to face all the difficulties he or she is going to face in the operating room. Other features (such as patient-specific constitute behavior of soft tissues) could be obtained by elastography, for instance, giving rise to a fully personalized patient avatar. This constitutes our current effort of research.

5 Acknowledgments

This work has been partially funded by the Spanish Ministry of Economy and Innovation through Grant number DPI2014-51844-C2-1-R. This support is gratefully acknowledged.

Collaboration provided by S. Nicolau, from IRCAD, France, is also gratefully acknowl-

edged.

References

[1] Iciar Alfaro, David Gonzalez, Felipe Bordeu, Adrien Leygue, Amine Ammar, Elias Cueto, and Francisco Chinesta. Real-time in silico experiments on gene regulatory networks and surgery simulation on handheld devices. *J. Comput. Surg.*, 1(1):2194–3990, 2014.

[2] Grégoire Allaire, François Jouve, and Anca-Maria Toader. Structural optimization using sensitivity analysis and a level-set method. *J. computational physics*, 194(1):363–393, 2004.

[3] A. Ammar, F. Chinesta, P. Diez, and A. Huerta. An error estimator for separated representations of highly multidimensional models. *Comput. Methods Appl. Mech. Eng.*, 199(25-28):1872 – 1880, 2010.

[4] A. Ammar, E. Cueto, and F. Chinesta. Reduction of the chemical master equation for gene regulatory networks using proper generalized decompositions. *Int. J. for Numer. Methods Biomed. Eng.*, in press, 2012.

[5] C. Bernoulli. *Vademecum des Mechanikers*. Cotta, 1836.

[6] M. Bro-Nielsen and S. Cotin. Real-time volumetric deformable models for surgery simulation using finite elements and condensation. *Comput. Graph. Forum*, 15(3):57–66, 1996.

[7] Martin Burger, Benjamin Hackl, and Wolfgang Ring. Incorporating topological derivatives into level set methods. *J. Comput. Phys.*, 194(1):344–362, 2004.

[8] F. Chinesta, A. Ammar, and E. Cueto. Recent advances in the use of the Proper Generalized Decomposition for solving multidimensional models. *Arch. Comput. Methods Eng.*, 17(4):327–350, 2010.

[9] F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta. PGD-Based Computational Vademecum for Efficient Design, Optimization and Control. *Arch. Comput. Methods Eng.*, 20(1):31–59, 2013.

[10] Francisco Chinesta and Elias Cueto. *PGD-Based Modeling of Materials, Structures and Processes*. Springer International Publishing Switzerland, 2014.

[11] Francisco Chinesta, Pierre Ladeveze, and Elias Cueto. A short review on model order reduction based on proper generalized decomposition. *Arch. Comput. Methods Eng.*, 18:395–404, 2011.

[12] Stéphane Cotin, Hervé Delingette, and Nicholas Ayache. Real-time elastic deformations of soft tissues for surgery simulation. In Hans Hagen, editor, *IEEE Transactions on Visualization and Computer Graphics*, volume 5 (1), pages 62–73. IEEE Computer Society, 1999.

[13] Hadrien Courtecuisse, Jérémie Allard, Pierre Kerfriden, Stéphane P.A. Bordas, Stéphane Cotin, and Christian Duriez. Real-time simulation of contact and cutting of heterogeneous soft-tissues. *Med. Image Analysis*, 18(2):394 – 410, 2014.

[14] Hadrien Courtecuisse, Hoeryong Jung, Jérémie Allard, Christian Duriez, Doo Yong Lee, and Stéphane Cotin. Gpu-based real-time soft tissue deformation with cutting and haptic feedback. *Prog. Biophys. Mol. Biol.*, 103(2-3):159 – 168, 2010. Special Issue on Biomechanical Modelling of Soft Tissue Motion.

[15] Elias Cueto and Francisco Chinesta. Real time simulation for computational surgery: a review. *Adv. Model. Simul. Eng. Sci.*, 1(1):11, 2014.

[16] H. Delingette and N. Ayache. Soft tissue modeling for surgery simulation. In N. Ayache, editor, *Computational Models for the Human Body*, Handbook of Numerical Analysis (Ph. Ciarlet, Ed.), pages 453–550. Elsevier, 2004.

[17] Herve Delingette and Nicholas Ayache. Hepatic surgery simulation. *Commun. ACM*, 48:31–36, February 2005.

[18] D.V. Dimitrov. Systems patientomics: The virtual in-silico patient. *New Horizons Transl. Medicine*, 2(1):1 – 4, 2014.

[19] David Gonzalez, Elias Cueto, and Francisco Chinesta. Real-time direct integration of reduced solid dynamics equations. *Internatinal J. for Numer. Methods Eng.*, accepted, 2014.

[20] R. B. Laughlin and David Pines. The theory of everything. *Proc. Natl. Acad. Sci.*, 97(1):28–31, 2000.

[21] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4):163–169, August 1987.

[22] U. Meier, O. Lopez, C. Monserrat, M. C. Juan, and M. Alcaniz. Real-time deformable models for surgery simulation: a survey. *Comput. Methods Programs Biomed.*, 77(3):183–197, 2005.

[23] Karol Miller, Grand Joldes, Dane Lance, and Adam Wittek. Total lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. *Commun. Numer. Methods Eng.*, 23(2):121–134, 2007.

[24] David Modesto, Sergio Zlotnik, and Antonio Huerta. Proper Generalized Decomposition for parameterized helmholtz problems in heterogeneous and unbounded domains: application to harbor agitation. *Comput. Methods Appl. Mech. Eng.*, 2015.

[25] S. Niroomandi, D. González, I. Alfaro, F. Bordeu, A. Leygue, E. Cueto, and F. Chinesta. Real-time simulation of biological soft tissues: a PGD approach. *Int. J. for Numer. Methods Biomed. Eng.*, 29(5):586–600, 2013.

[26] S. Niroomandi, D. Gonzalez, I. Alfaro, E. Cueto, and F. Chinesta. Model order reduction in hyperelasticity: a proper generalized decomposition approach. *Int. J. for Numer. Methods Eng.*, 96(3):129–149, 2013.

[27] Marzia Polito and Pietro Perona. Grouping and dimensionality reduction by locally linear embedding. In *Advances in Neural Information Processing Systems 14*, pages 1255–1262. MIT PRess, 2001.

[28] Guenhael Le Quilliec, Balaji Raghavan, and Piotr Breitkopf. A manifold learning-based reduced order model for springback shape characterization and optimization in sheet metal forming. *Comput. Methods Appl. Mech. Eng.*, 285(0):621 – 638, 2015.

[29] Balaji Raghavan, Liang Xia, Piotr Breitkopf, Alain Rassineux, and Pierre Villon. Towards simultaneous reduction of both input and output spaces for interactive simulation-based structural design. *Comput. Methods Appl. Mech. Eng.*, 265(0):174 – 185, 2013.

[30] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Sci.*, 290(5500):2323–2326, 2000.

[31] S. Schuon, M. Durkovic, K. Diepold, J. Scheuerle, and S. Markwardt. Truly Incremental Locally Linear Embedding. In *Proceedings of the CoTeSys 1st International Workshop on Cognition for Technical Systems*, October 2008.

[32] Neeraj. Sharma and Lalit. Aggarwal. Automated medical image segmentation techniques. *J. Med. Phys.*, 35(1):3–14, 2010.

[33] Z.A. Taylor, M. Cheng, and S. Ourselin. High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. *Med. Imaging, IEEE Transactions on*, 27(5):650 –663, may 2008.

List of Figures

Figure 1: Prototype of surgery simulator based on the use of a computational vademecum. Palpation of the liver during cholecystectomy is studied.

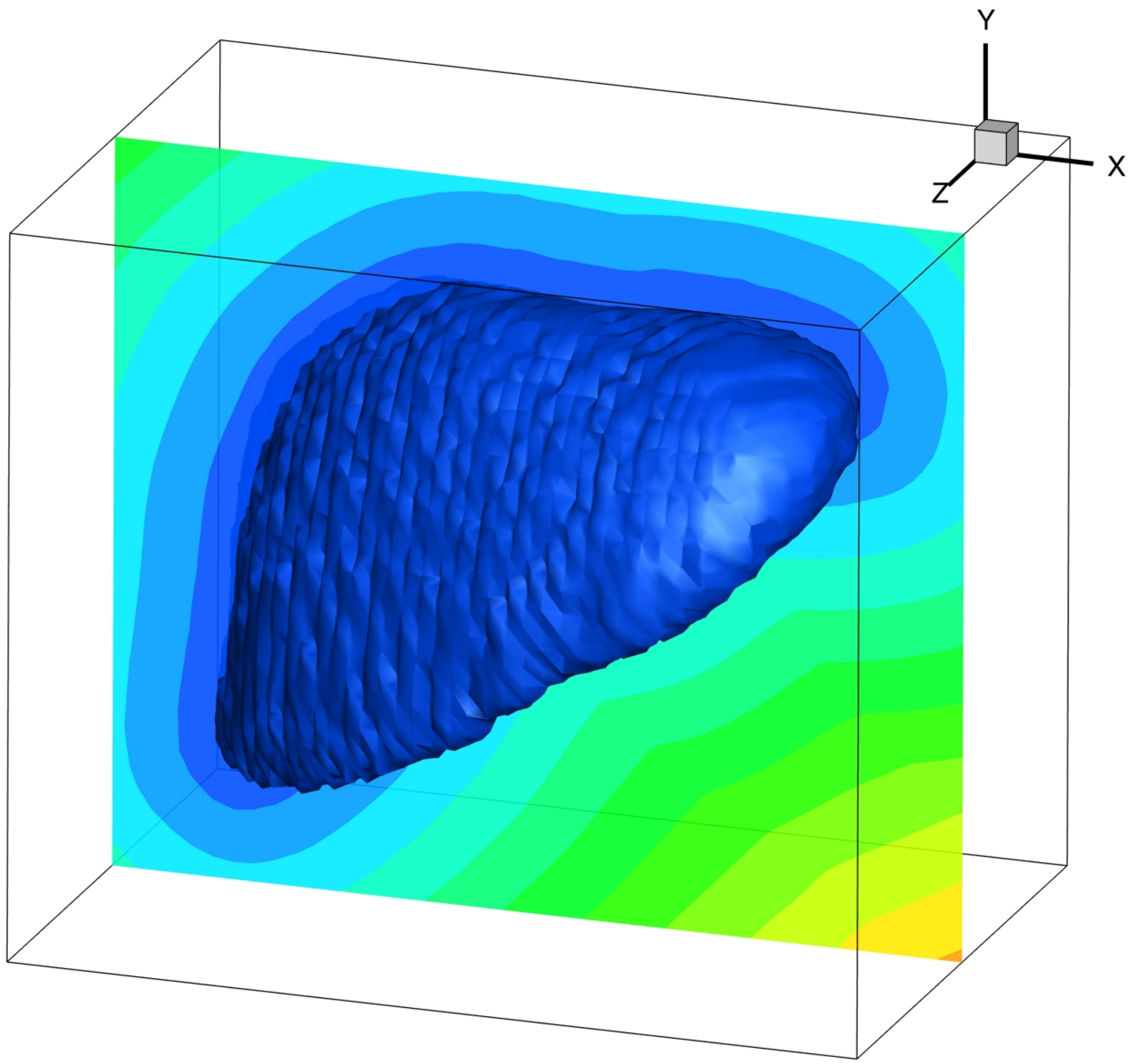


Figure 2: Distance field for a particular liver and associated geometry extracted by the Marching Cubes algorithm [21].

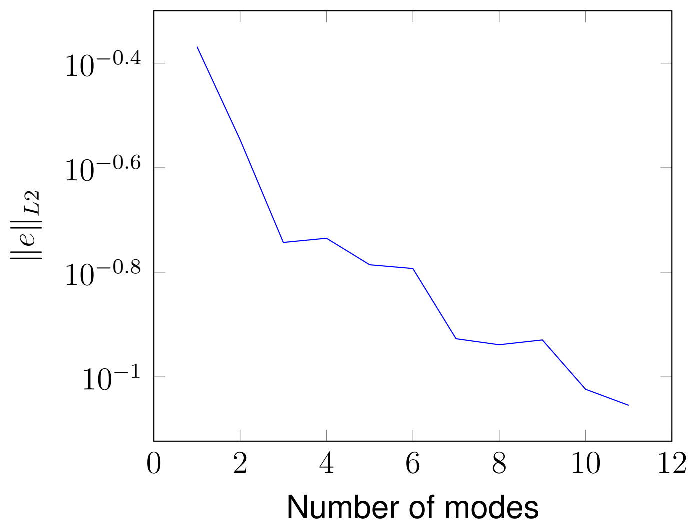


Figure 3: Convergence of the PGD approximation of a single vademecum towards the reference FEM solution for different number of modes.

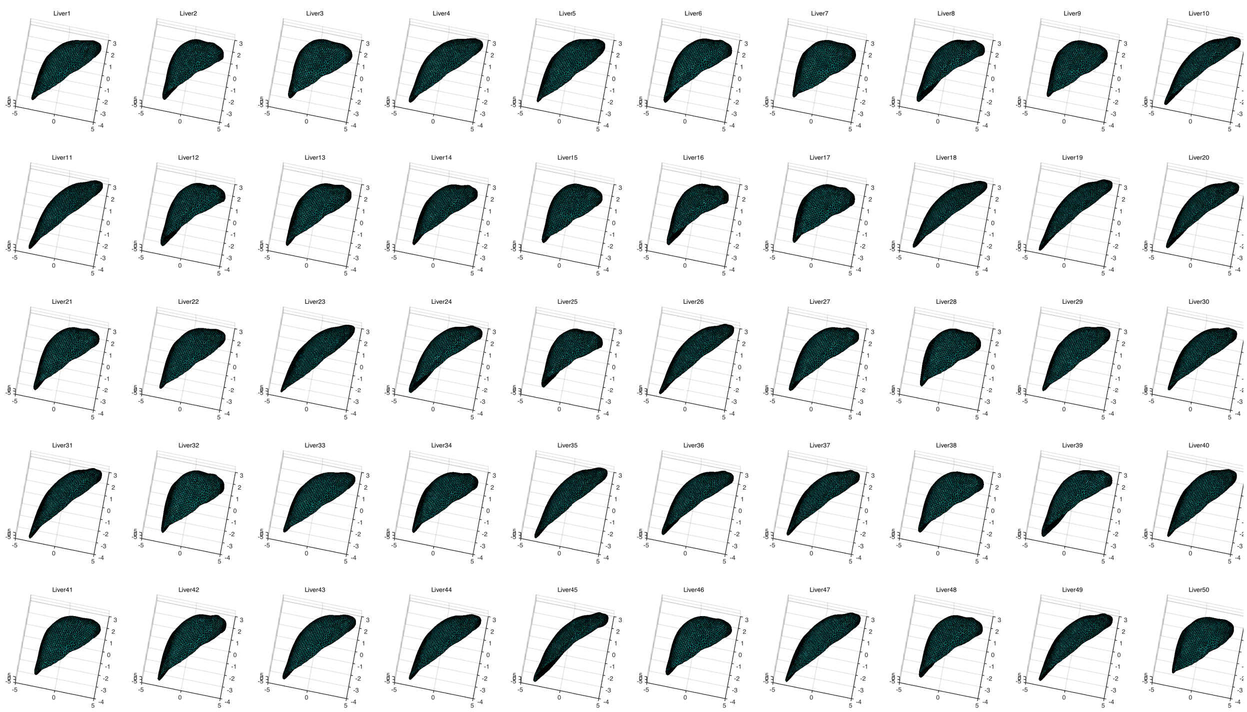
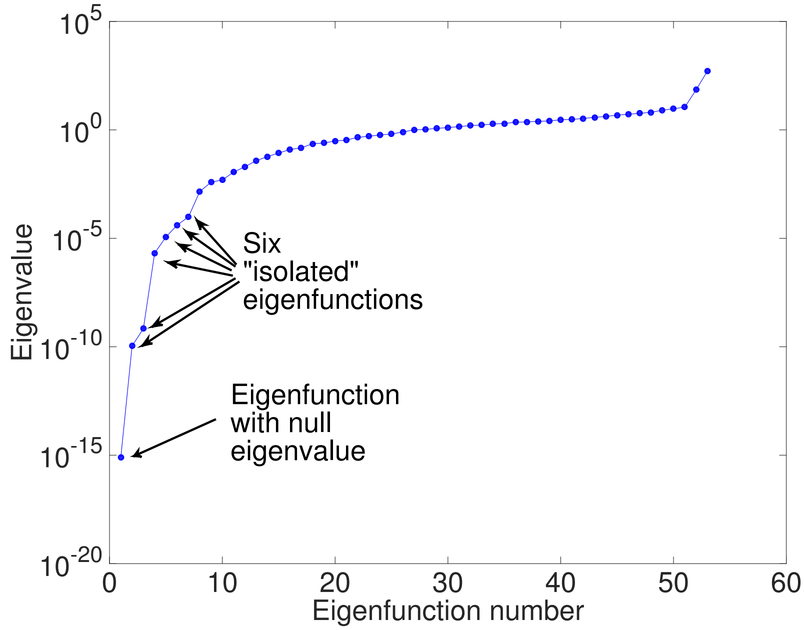
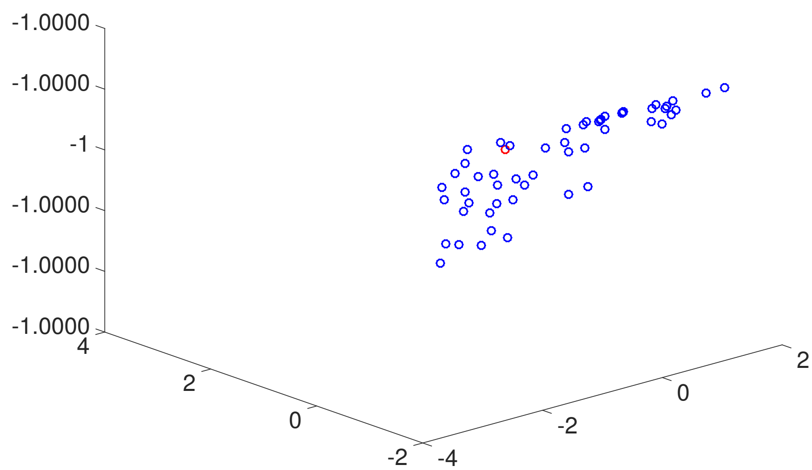


Figure 4: Set of 50 different livers generated by applying a linear transformation to an original liver geometry.



(a)



(b)

Figure 5: (a) Set of predicted eigenvalues, showing the 6-dimensional structure of the set of livers. (b) Result of the projection into a 3-d space, where the flat structure of the set can be noticed. The red dot represents the liver to be interpolated.

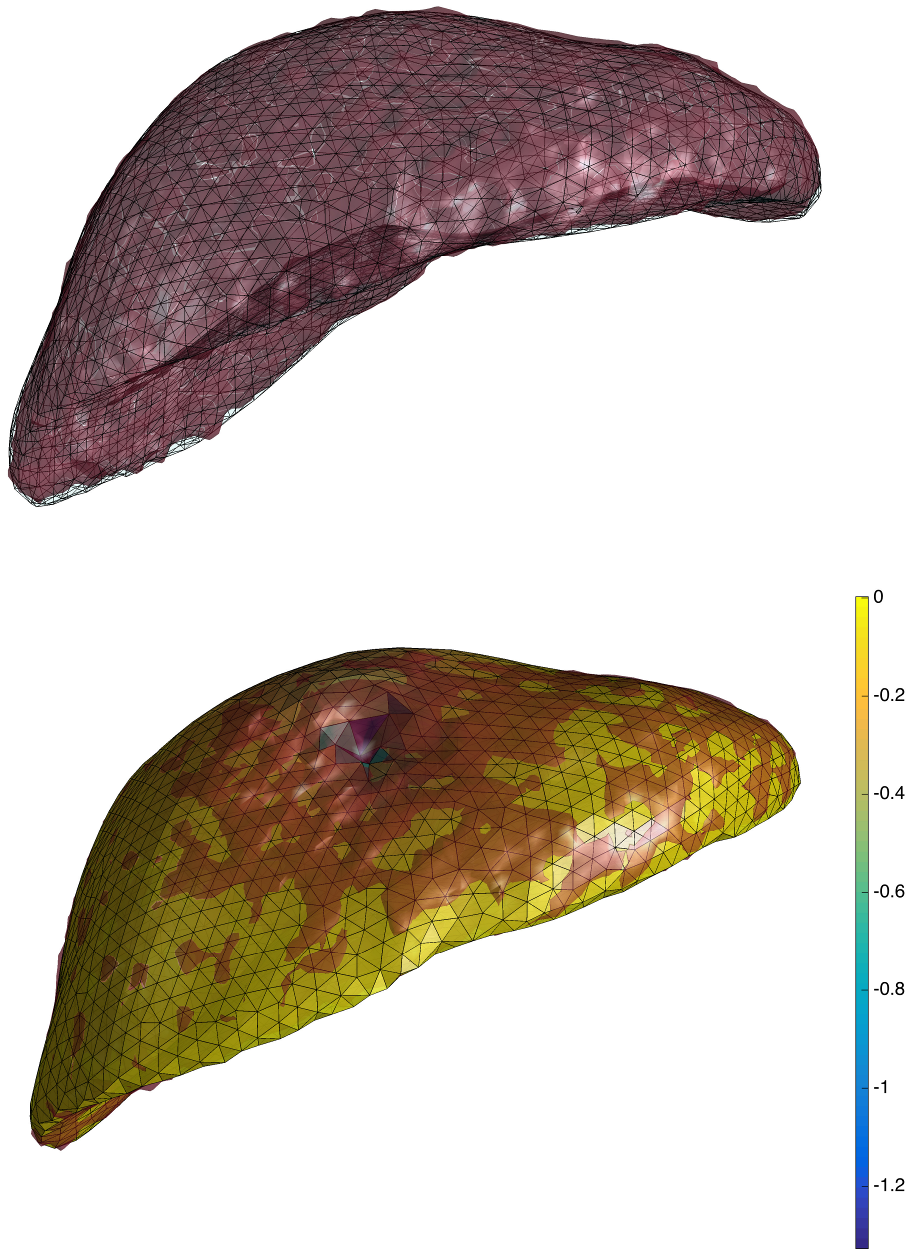


Figure 6: (a) Comparison between an interpolated anatomy and the original one. Reference anatomy is represented in solid, while the obtained interpolation is represented in wireframe. (b) Comparison between an interpolated vademecum and the original one. Both deformed geometries are superimposed to show the noticeable similarity between them. The color bar represents the magnitude of the difference in displacement, in mm.

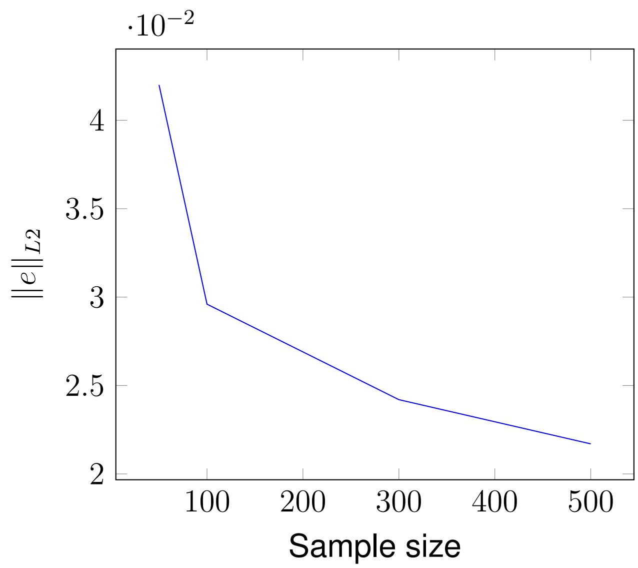


Figure 7: Convergence rate in the level set approximation of one particular liver of the set, measured in $L2$-norm.

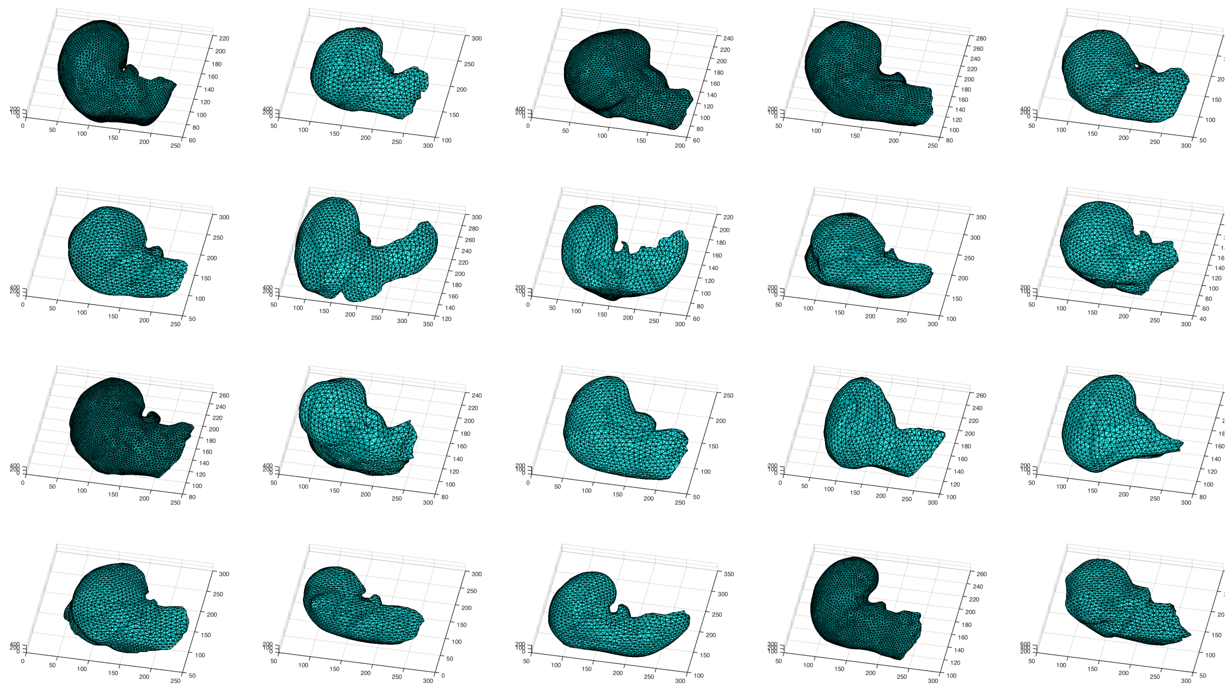


Figure 8: Geometry of the 20 samples of patient livers.

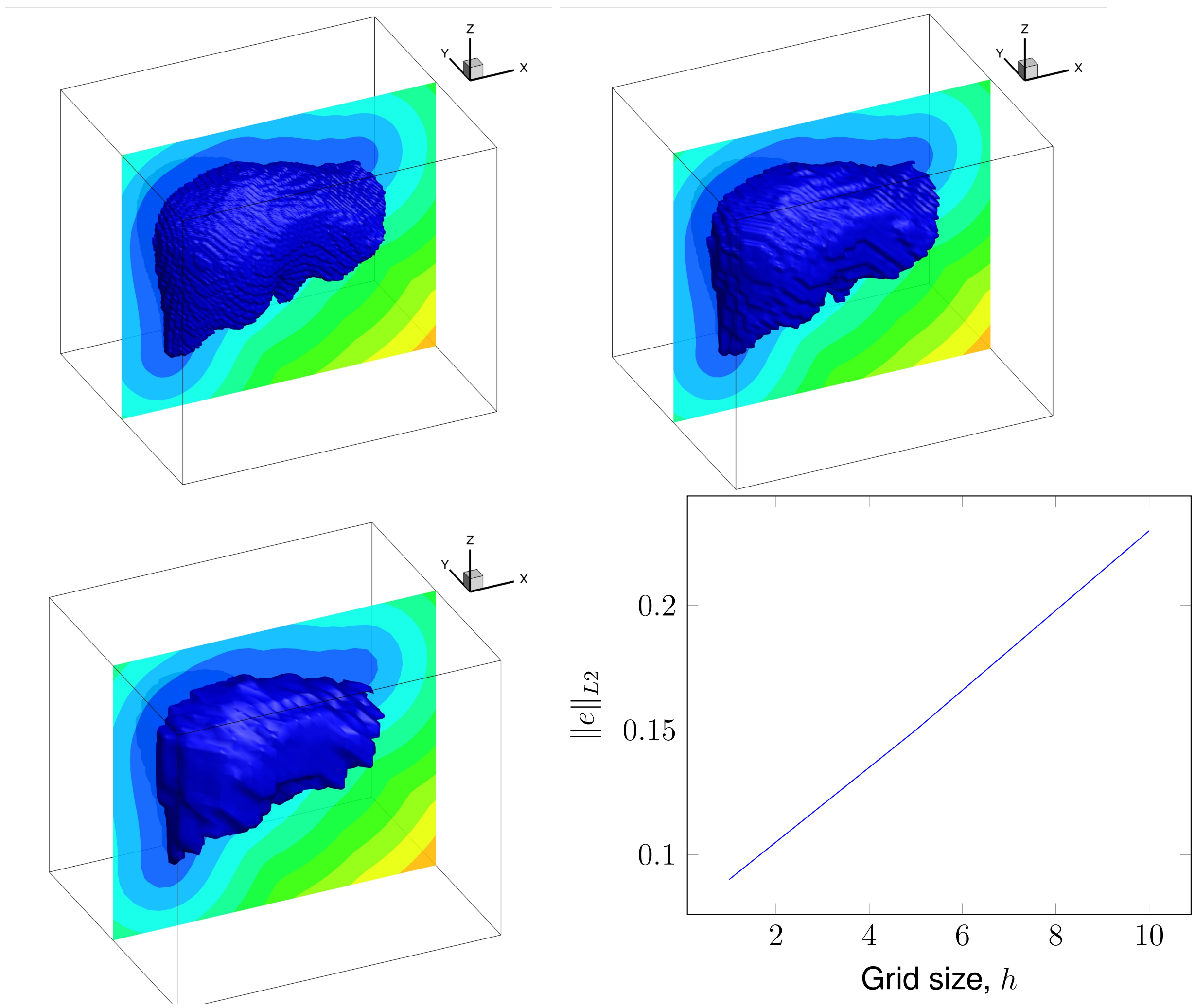


Figure 9: Obtained anatomies for different level set grid sizes: (a) 1 mm, (b) 5 mm, (c) 10 mm. (d) Convergence of the predicted volume (L_2-norm error of the distance field within the liver geometry) of the three different livers with grid size.

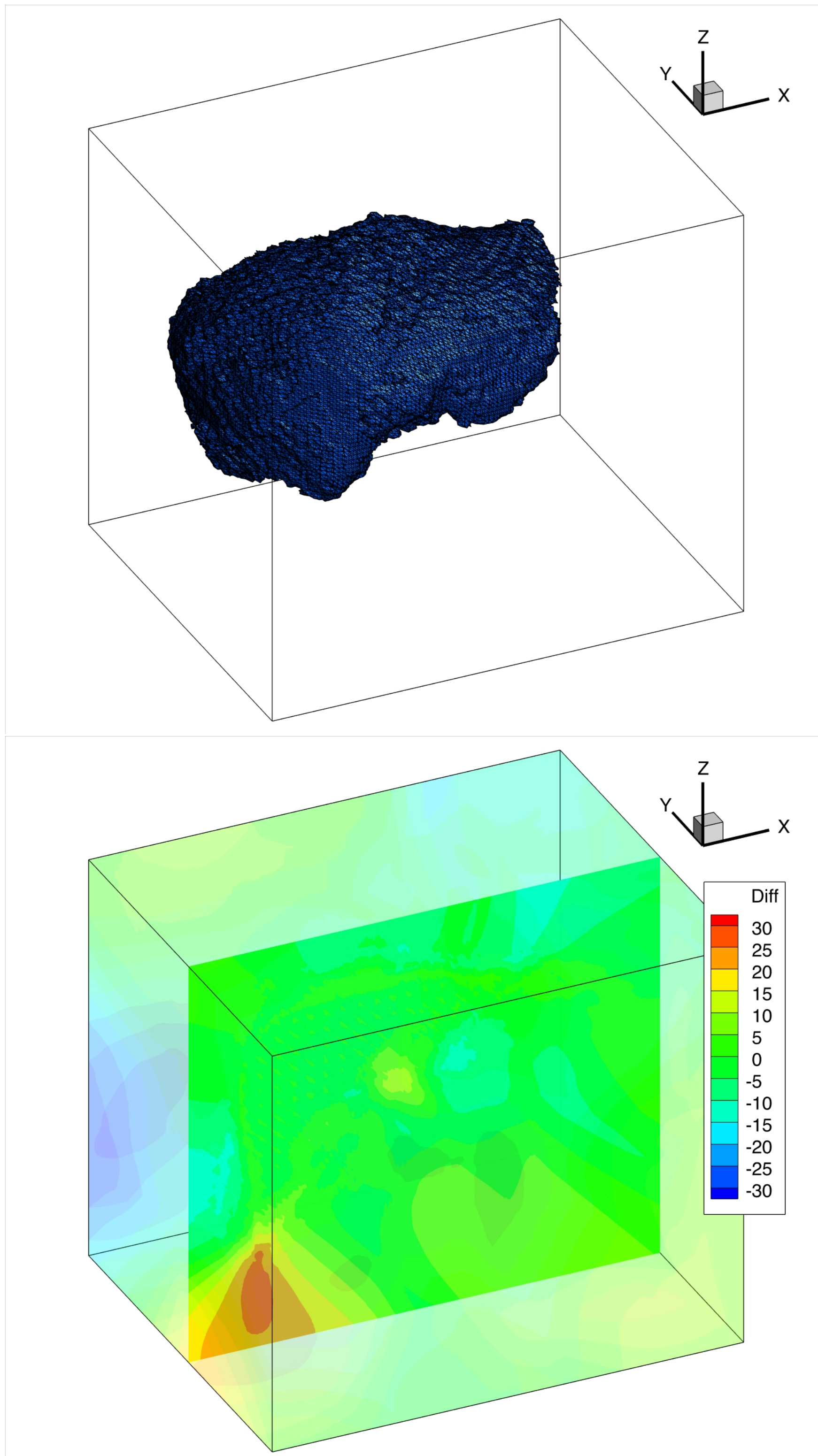


Figure 10: Obtained anatomy for liver number 13 (a). Error with respect to the target anatomy measured as the difference between distance real and interpolated fields (b).