A model order reduction approach to create patient-specific mechanical models of human liver in computational medicine applications.

Nathan Lauzeral, Domenico Borzacchiello, Michael Kugler, Yves Rémond, Daniel George, Alexandre Hostettler, Francisco Chinesta

▶ **To cite this version:**

Nathan Lauzeral, Domenico Borzacchiello, Michael Kugler, Yves Rémond, Daniel George, et al.. A model order reduction approach to create patient-specific mechanical models of human liver in computational medicine applications.. Computer Methods and Programs in Biomedicine, 2019, 170, pp.95-106. 10.1016/j.cmpb.2019.01.003 . hal-02063517

HAL Id: hal-02063517
https://hal.science/hal-02063517

Submitted on 11 Mar 2019

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

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

A model order reduction approach to create patient-specific mechanical models of human liver in computational medicine applications

Nathan Lauzeral[a,*], Domenico Borzacchiello[a], Michael Kugler[b,c], Daniel George[b], Yves Rémond[b], Alexandre Hostettler[c], Francisco Chinesta[d]

[a] *ICI, High Performance Computing Institute, Ecole Centrale de Nantes, France*
[b] *iCube, Université de Strasbourg, CNRS, France*
[c] *IRCAD, France*
[d] *PIMM, ENSAM ParisTech, CNRS, France*

ARTICLE INFO

Keywords:
Real-time simulation
Patient-specific modeling
Data-based modeling
Statistical shape analysis
Finite element modeling
Human liver

ABSTRACT

Background and objective: This paper focuses on computer simulation aspects of Digital Twin models in the medical framework. In particular, it addresses the need of fast and accurate simulators for the mechanical response at tissue and organ scale and the capability of integrating patient-specific anatomy from medical images to pinpoint the individual variations from standard anatomical models.

Methods: We propose an automated procedure to create mechanical models of the human liver with patient-specific geometry and real time capabilities. The method hinges on the use of Statistical Shape Analysis to extract the relevant anatomical features from a database of medical images and Model Order Reduction to compute an explicit parametric solution for the mechanical response as a function of such features. The Sparse Subspace Learning, coupled with a Finite Element solver, was chosen to create low-rank solutions using a non-intrusive sparse sampling of the feature space.

Results: In the application presented in the paper, the statistical shape model was trained on a database of 385 three dimensional liver shapes, extracted from medical images, in order to create a parametrized representation of the liver anatomy. This parametrization and an additional parameter describing the breathing motion in linear elasticity were then used as input in the reduced order model. Results show a consistent agreement with the high fidelity Finite Element models built from liver images that were excluded from the training dataset. However, we evidence in the discussion the difficulty of having compact shape parametrizations arising from the extreme variability of the shapes found in the dataset and we propose potential strategies to tackle this issue.

Conclusions: A method to represent patient-specific real-time liver deformations during breathing is proposed in linear elasticity. Since the proposed method does not require any adaptation to the direct Finite Element solver used in the training phase, the procedure can be easily extended to more complex non-linear constitutive behaviors - such as hyperelasticity - and more general load cases. Therefore it can be integrated with little intrusiveness to generic simulation software including more sophisticated and realistic models.

1. Introduction

For the sake of clarity, abbreviations and main mathematical notations are summarized in Tables 1 and 2.

1.1. Medical context

Thanks to recent advances in diverse medical technologies, augmented reality (AR) is now on the doorsteps of operating rooms (OR). Such technology opens up many perspectives for various medical applications such as interventional radiology (IR) or minimally invasive surgery (MIS) [2,32]. The main idea is to display

* Corresponding author.
E-mail address: nathan.lauzeral@ec-nantes.fr (N. Lauzeral).

Table 1
List of abbreviations.

AR	Augmented Reality
CT	Computed Tomography
FE	Finite Element
iFEMWRAP	Iterative FEMWRAP [43]
IR	Interventional Radiology
MIS	Minimally Invasive Surgery
MOR	Model Order Reduction
MRI	Magnetic Resonance Image
OR	Operating Room
PCA	Principal Component Analysis
PDM	Point Distribution Model
PGD	Proper General Decomposition
pROM	Parametric Reduced Order Model
ROM	Reduced Order Model
SSA	Statistical Shape Analysis
SSL	Sparse Subspace Learning
SSM	Statistical Shape Model
TPS	Thin Plate Spline
TPS-PR	Thin Plate Spline Parametrized Registration
VOE	Volumetric Overlap Error

Table 2
List of notations.

T	Number of liver shapes in the database.
v_V	Number of nodes in the liver template mesh.
$\boldsymbol{u}$	Displacement field associated to the liver mesh.
$\boldsymbol{u}_d$	User-prescribed motion on the surface of the liver to represent the breathing motion.
$\boldsymbol{\xi}$	Vector containing the implicit parameters ($\boldsymbol{\xi} = (\xi_1, \cdots, \xi_{n_p}) \in \mathbb{R}^{n_p}$). In the application of this paper they represent the liver's shape.
$\boldsymbol{p}$	Vector containing the explicit parameters ($\boldsymbol{p} = (p_1, \cdots, p_D) \in \mathbb{R}^D$).
b	Breathing parameter ($b \in [0, 1]$).
$\bar{\boldsymbol{X}}$	Liver shape parametrization.
n_p	Number of shape parameters used to described a liver shape.
n_b	Discretization of the regular grid containing the control points used in the TPS-PR algorithm ($n_b \in \mathbb{N}^*\backslash\{1\}$), see Appendix A.
λ_r	Tuning factor used in the TPS-PR algorithm, see Appendix A.
M	Number of modes used for the shape reconstruction in the different error measures, see Section 2.6.

virtual information on real images of the patient to see through opaque tissues. In IR, this augmented representation of the patient is important to visually follow the targeted organs. For example, in radiotherapy such tool allows to follow the tumor and predict its displacement during the breathing [21], permitting to minimize the radiated area and consequently the trauma for the patient. Concerning the MIS, the AR is used as a back up for two major drawbacks of the technique: the reduced visibility and in the case of robot-assisted MIS the loss of haptic feedback during the operation [38]. Thanks to AR, it is now possible to provide the surgeon with additional information such as the position of tumors [15,16] or blood vessels [11,15]. Modifications in the organ topology due to cuts can even be taken into account [40]. Results on haptic feedback technologies in AR, although not yet robust enough to be used in the ORs, are showing promising results and are currently used as training and learning tools [39,45]. However, these approaches are often more challenging when applied to soft tissues because of the difficulty to continuously adapt the AR scene to the position of the features of interest. In abdominal surgery, for example, a lot of factors can interact with the organs. They can come from the internal or external environment (surgical tools, insufflation of gas in the peritoneum) or be physiological movements (heart beats, breathing). Hence the necessity to develop robust real-time biomechanical models of the tissues to integrate these movements and reproduce them with fidelity.

1.2. Scope of the current work

In most computational medicine applications the goal is to create a digital replica of a considered biophysical system that can realistically reproduce the most essential observed features. Two key aspects in this are the possibility to customize the models for specific use cases (accounting for inter-patient and inter-population variabilities) and the real-time interactive response to new assimilated data. This emerging technology, known as the Digital Twin, merges complex biophysical modeling and advanced real-time simulation techniques with data assimilation and analysis for decision support. Some model and simulation aspects were addressed in the review by [10] in the surgical framework and the use of machine learning is investigated in [26,29,30] for parameters identification and mechanical behavior prediction. More specifically, models for visual and haptic feedack applications were developed in [34,35,37], where the non-linearities and load parametrization issues are tackled, and in [36,42], where the simulation of surgical cutting is dealt with. In this paper we build on these works and focus on the aspects related to the integration of medical images in real-time interactive models to personalize the organ anatomy. We propose a numerical framework to personalize biomechanical models interactively using new anatomical data without rebuilding the models from scratch.

The specific anatomy of the patient is generally taken into account case-by-case thanks to pre-operative data such magnetic resonance images (MRI) or computed tomography (CT) scans. From these inputs, the surface and the volume of the organs of interest can be identified, reconstructed and meshed [1,18,24,46]. This can be a time-consuming and computationally intensive task and needs to be repeated for any new patient, posing severe limitations to the use of such models within interactive simulation environments. Then, to reach real-time performance the models either need to simplify the formulation, e.g. using lumped parameters models, corotational finite elements or beam structures [15], or use specific hardware set-up for fast parallel computations [22]. In the first case, over simplification in the model formulation may lead to incorrect mechanical behavior unless the lumped parameters are accurately tuned to capture the essential features of the equivalent distributed parameters model. In the second one, it involves high performance computing resources that are rarely available in a clinical environment. In both cases, computational time will strongly depend on the mesh refinement, limiting the accuracy to the hardware capacity. In this paper we follow the route of collocation-based Model Order Reduction (MOR). This approach allows to simultaneously tackle the specific anatomical representation and the real-time constraint without being limited by the simplification of the physics or the computational cost and bypassing mesh generation and model assembly costs. The reduction of computational costs also enables the use of inexpensive and simple hardware such as tablets or even smartphones, which could be easily installed in the ORs.

1.3. Reduced order modeling for organ twins models

MOR methods have seen a growing interest this last decade. They allow to reduce the computational complexity in numerical simulations by a parametrization of the solution, enabling real-time *online* computations whithout simplifying the underlying physics of the model. In return, a computationally intensive *offline* stage must be done beforehand. They are good candidates to the creation of Digital Twins because they handle data assimilation through their parametric formulation. A distinction can be done between the projection-based and the collocation-based MOR methods. The underlying idea in the former is to seek the solution for the model's governing equations in a lower-dimensional

subspace that is specific for the problem at hand. This is accomplished by imposing the orthogonality of the equations residuals with respect to the new subspace. The resulting system of equations has a considerably reduced computational complexity, while the accuracy of the solutions is preserved. Based on the way the reduced basis is *learned*, projection-based MOR methods are usually divided in two categories: the *a posteriori* methods like the Proper Orthogonal Decomposition [7] or Reduced Basis method [41], where the reduced order model (ROM) is built from a set of training solutions, available from previous simulations (called snapshots), and the *a priori* methods like the Proper Generalized Decomposition (PGD) [8], in which the parametric solution is assumed to respect the canonical tensor format and is then built from the equations governing the problem. These methods have been used for different applications in relation with the medical framework, and in particular computational surgery [10]. Projection-based ROM including geometrical parameters have been addressed in [28] for optimization purposes and in [13] to synthesize new shapes from manifold interpolation. When using real medical data, the first difficulty to create anatomical models encompassing the shape comes from its parametrization. This latter can be obtained through the use of Statistical Shape Models (SSMs), and, in particular, Point Distribution Models (PDMs) [9]. From a training set of real anatomical shapes the PDMs associate to a template the principal modes of shape variation. This way, the geometry of a specified organ can be statistically described by a limited set of parameters allowing image segmentation [17] or the creation of Finite Element (FE) shape parametrized models [5,25,33].

A key assumption for projection-based ROM is that the problem variational form is affine with respect to the parameters. This guarantees the complexity reduction in the projection step and is therefore fundamental for the performance of the method in terms of execution time. Unfortunately, affine approximations are not always easily recovered. This bottleneck is most likely to be encountered in problems involving a parametrization of the domain geometry, as in our case. To efficiently handle the representation of the shape, we opted for a collocation-based MOR method, the Sparse Subspace Learning (SSL) [4]. On the contrary to projection-based approaches, in collocation-based methods the residual is not constrained to be orthogonal to the solution's subspace but rather enforced to be exactly zero at properly chosen points in the parametric space, called the collocation points. The parametric solution is built by interpolating the solutions of the model at these points, which makes the ROM an *a posteriori* model. In our case, the main advantage is that the approach is fully data-driven, as it does not require the evaluation of the residual nor its projection onto the lower dimensional space. Hence, the lack of affinity of the variational form does not affect the performance of the method. As the shape parametrization is based on a PDM which results in the loss of affinity, it motivates our choice of the SSL. This latter is based on the use of hierarchical collocation to build the parametric model, exploiting sparsity and low-rank representation for the solutions. SSL aims for the sparsity of the representation using a sparse grid sampling approach while simultaneously building a reduced order representation of the solution. New out-of-training solutions can be computed by a hierarchical interpolation of the snapshots coefficients in the reduced basis. Since SSL is only based on the output of a FE solver and does not require any modification of it, it is regarded as a non-intrusive technique. However, the use of hierarchical collocation in the parametric space, in our case hierarchical polynomials, requires a fundamental condition on the regularity of the solution in the parametric space of shape features. To preserve this regularity, we adopted a morphing technique to smoothly adapt the FE mesh of the model according to the shape parameters [25].

1.4. Previous work

In our previous paper [25] we developed a method to build shape-parametrized FE models from PDMs through a morphing technique. The main idea is to create the SSMs in such way that the generated shapes are adapted to the iFEMWRAP method [44] used to reconstruct the volumetric mesh. This latter enables the reconstruction of a 3D mesh from the knowledge of an initial state plus displacements applied to the boundary nodes, i.e. the external mesh's surface. By iteratively minimizing the energy of deformation of the volume the position of the inner nodes is computed in such way that the mesh is adapted to FE computations. The main constraint is that the imposed displacements on the surface do not generate a tangled surface mesh. To this end we introduced a registration algorithm called Thin Plate Spline Parametrized Registration (TPS-PR) (see Appendix A), which maximizes the morphing efficiency. By parametrizing the shape of a template 3D volumetric mesh through a Statistical Shape Analysis (SSA) using the TPS-PR algorithm, the iFEMWRAP can then be applied to any generated shape using the template as initial state. In the remainder of this paper this method is going to be enhanced and adapted to the SSL in order to create the parametric ROM (pROM) for breathing simulations, as an example application. This is motivated by previous works from Hostettler et al. [20,21], where a method to predict the abdominal organs and tumors positions during free breathing was developed.

1.5. Overview and paper organization

This paper presents a method based on the SSL to create patient-specific mechanical models of the human liver by taking into account their anatomy. To do so, a SSA is used in order to get the shape parametrization of the human liver, allowing to recreate the shape of any new patient. By combining the SSA and the SSL, one can solve in real-time a chosen mechanical problem on any shape taken into account by the parametrization. In this paper the specific case of the breathing is developed. During the breathing, the displacement of the liver's surface can be estimated. From this input the presented method allows to reconstruct the organ inner displacements accordingly to the material's constitutive behavior used to represent it, enabling the visualization of tumors or blood vessels positions within the liver. We first present the liver shape database used for the SSA in Section 2.1. The mechanical model is introduced in Section 2.2 and the associated breathing simulation is presented in Section 2.3. The method to create the pROM is described in Section 2.4, followed by the process to adapt the model to a new patient in Section 2.5. Then, error metrics are defined and generated results are presented and commented in Sections 2.6 and 3. Finally, the identified limits of the method are highlighted in the discussion and some potential solutions are suggested.

Data acquisition, processing and assimilation as well as more technological aspects are not covered in this work and will be object of a parallel publication.

2. Materials and methods

If not specified otherwise, all vectors mentioned by a bold italic letter belong to a three dimensional space (x, y, z). A point cloud $\boldsymbol{X} \in \mathbb{R}^{n\times 3}$ will correspond to $\boldsymbol{X} = (x_1, \cdots, x_n, y_1, \cdots, y_n, z_1, \cdots, z_n)^T$.

2.1. Data

A database of $T = 385$ external surface meshes of livers was provided by the IRCAD (Institut de Recherche contre les Cancers

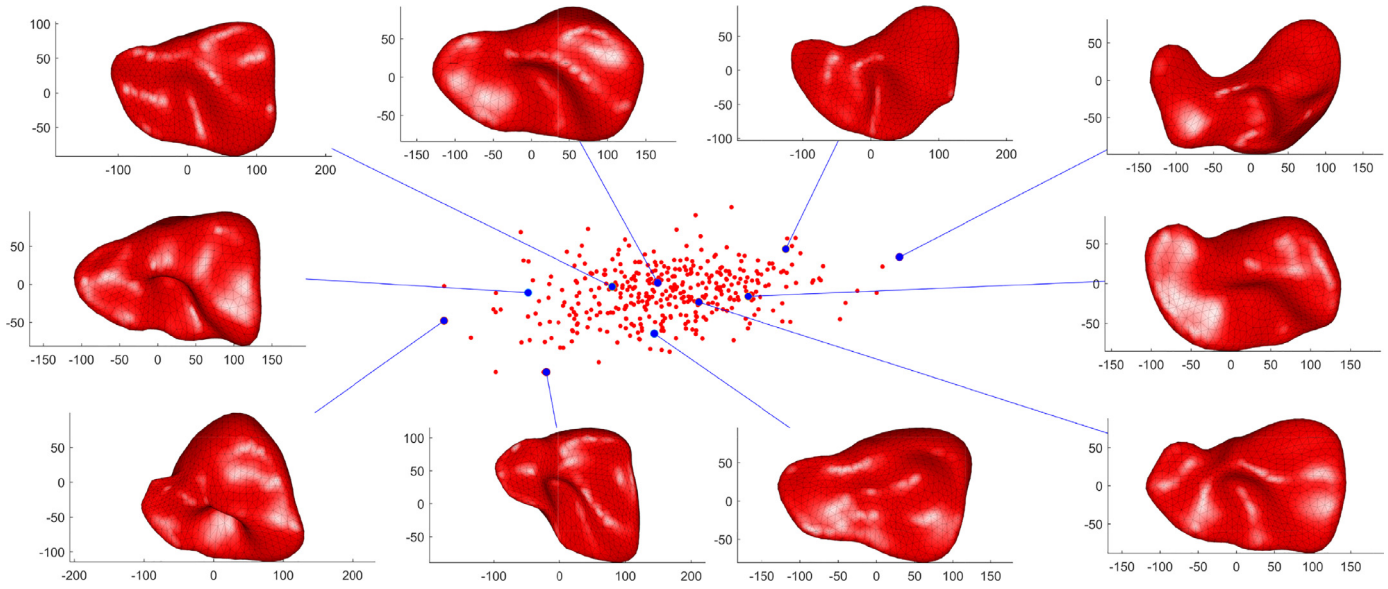


Fig. 1. 2D mapping of the liver shape population using PCA along the first and second principal axes. Some of the registered target shapes are displayed and linked to their two dimensional representations. Shapes taken from the dense area are similar, whereas isolated ones are unique. During the construction of the shape parameters boundaries (Appendix B and Fig. 5c) some of these shapes are excluded in order to generate a more precise and compact pROM. The visualization was done on Matlab.

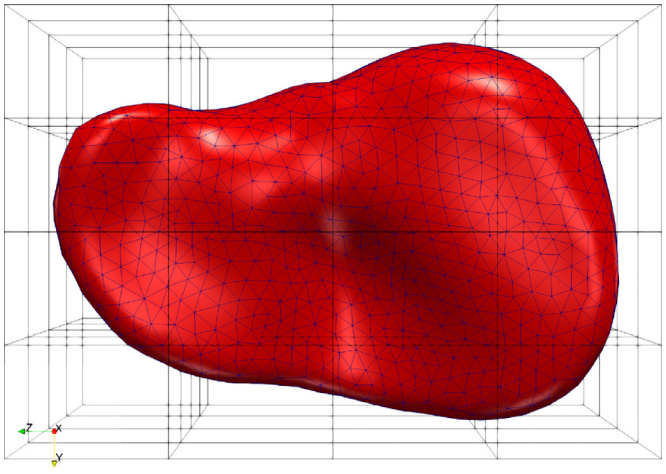


Fig. 2. Template liver shape. The control points associated to the spline representation used in the non-rigid registration method (see Appendix A) are shown, the discretization used is $n_b = 5$. The mesh has been generated using gmsh and the display is done on Paraview.

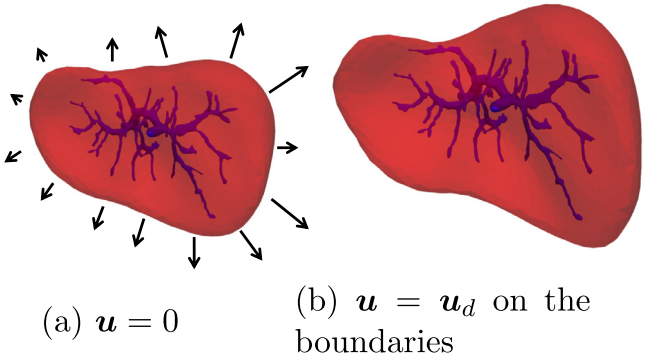


Fig. 3. Visualization of the liver (in red) with the representation of the portal vein (in blue) for different values of $\boldsymbol{u}$. The black arrows represent the direction of $\boldsymbol{u}_d$. One of the interest of the procedure is to be able to display internal features to see through the opaque tissues of the liver. The position of the portal vein is deduced from the liver displacement fields computed through the mechanical equilibrium laws (Eq. (1)).

de l'Appareil Digestif, Strasbourg, France). To our knowledge this is one of the biggest and diverse database used in the literature to build SSMs on a human organ. This database contains a large scale of liver shape collected on various kind of patients (male or female, healthy or unhealthy) in supine position. The shape meshes have different number of vertices (from 2000 to 150,000). Due to the fact that this database has been collected over many years, different qualities of segmentation are included. Some of the shapes are represented in Fig. 1.

A template shape close to the average target is chosen as reference (see Fig. 2) and meshed with gmsh [12]. Its surface is described by $v_S = 1393$ vertices and 2782 triangles. The volume is meshed with 10,163 tetrahedrons for a total of $v_V = 2452$ vertices. Its maximum length is 260mm. Special attention has been paid to the mesh quality using mesh optimization functions implemented in the open software gmsh.

2.2. Model assumptions

The pROM of the liver is created under the assumption of the quasi-static mechanical equilibrium with a prescribed displacement on the boundaries. The governing equations read:

$$\begin{cases} \nabla \cdot \sigma = 0 & \text{in the volume} \\ \boldsymbol{u} = \boldsymbol{u}_d & \text{on the boundaries} \end{cases} \tag{1}$$

where $\boldsymbol{u}$ is the displacement field and σ the Cauchy stress tensor. The user-prescribed motion $\boldsymbol{u}_d$ can be deduced from the position of the abdomen and that of the surrounding organs. For example, for an intergration in a complete AR tool one could use the work presented in [21] where the position of inner organs is estimated from the abdomen surface motion parametrization under free breathing. The stress-strain relationship is assumed to be linear and the material properties are considered homogeneous and isotropic, although, hypereleastic behavior could also be considered. Fig. 3 shows an example of geometry in which the

vascularization position is updated based on the computed displacement field.

2.3. Breathing simulation application

The liver is considered as an elastic solid defined by $E = 3kPa$ and $\nu = 0.48$. Its movement during the breathing is modeled by applying fixed Dirichlet boundary conditions to the surface nodes:

$$\boldsymbol{u}(b) = b \times \boldsymbol{u}_d \quad \text{on the boundaries,} \tag{2}$$

where $b \in [0, 1]$ is the breathing parameter. The displacement $\boldsymbol{u}_d$ has been computed by taking the extreme positions of a real liver during the breathing in vivo, from data provided by the IRCAD. By registering the template shape to the maximum inhalation and exhalation positions using the TPS-PR algorithm, the displacement can be directly applied to the template mesh.

In this case a single parameter has been used to describe the breathing motion. Multiple parameters could be used as well possibly leading to a more accurate representation. The parametrization could then be obtained by applying a SSA to a database of images corresponding to maximum inhalation and exhalation positions of several patients.

2.4. Creation of the parametric reduced order model

In the general case, the offline procedure to build pROM can be separated in two main steps: the parametrization of the problem and the use of a separated variables representation to enable fast computations during the online stage. An illustration of the method is given in Fig. 4.

Problem parametrization. To take into account the variability of the model according to the fields of interest, a parametric approach is used. Each field of interest is described by one or several parameters. In this paper a distinction is done between the explicit and the implicit parameters (respectively noted $\boldsymbol{p} = (p_1, \dots, p_D)$ and $\boldsymbol{\xi} = (\xi_1, \cdots, \xi_{n_p})$). The firsts encompass all kind of parametrization where the parameters are directly deduced from the formulation of the fields of interest. For example, in the case of the material properties, the Young modulus or the Poisson ratio. The same can goes for boundary conditions. In the breathing application presented in this article $\boldsymbol{p} = b$ is deduced from Eq. (2). On the contrary, implicit parameters are "hidden" within the model and must be inferred through methods such as the SSA used to represent the liver's shape. This latter is detailed in Appendix A and Appendix B using the data presented in subsection 2.1, resulting in the shape parametrization described by Eq. (10) and the parameters subspace given by Eq. (12). Fig. 5a–c illustrate how the shape parameters subspace boundaries are defined.

Separated variables representation. Once the model is parametrized, the solution - here the displacement field - is expressed using a separated variables representation. More specifically, the canonical tensor format is used, which allows to overcome the exponential complexity of multi-parametric models. This reads:

$$\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{p}) = \sum_{i=1}^{d} \alpha_i \boldsymbol{B}_0^i(\boldsymbol{x}) \prod_{k=1}^{n_p} B_k^i(\xi_k) \prod_{l=1}^{D} B_{n_p+l}^i(p_l)\,. \tag{3}$$

This formulation expresses the multidimensional field $\boldsymbol{u}$ as the truncated modal expansion in which each mode is the product of lower dimensional functions B_j^i expressing the parameter dependence. d is called the canonical rank of $\boldsymbol{u}$. For the application considered in this paper, Eq. (3) becomes:

$$\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{\xi}, b) = \sum_{i=1}^{d} \alpha_i \boldsymbol{B}_0^i(\boldsymbol{x}) \prod_{k=1}^{n_p} B_k^i(\xi_k) B_{n_p+1}^i(b)\,. \tag{4}$$

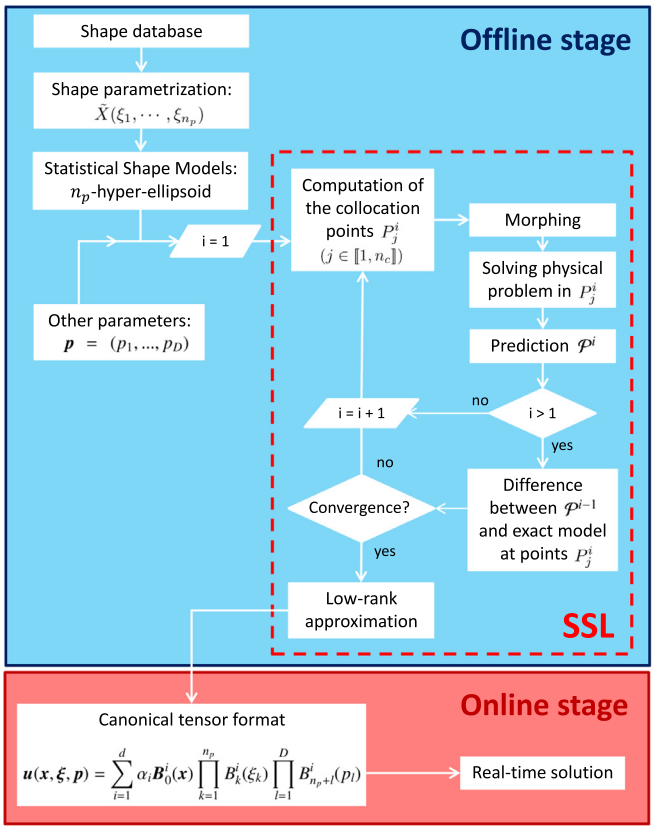
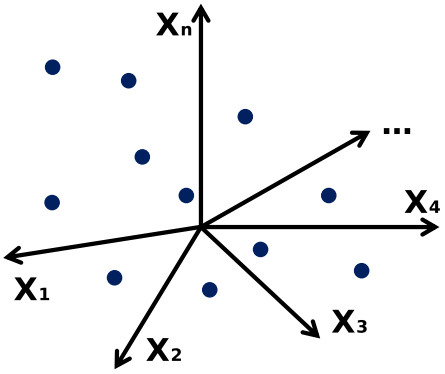


Fig. 4. Summary of the method to construct the patient-specific pROM. First an *offline* stage is done where the shape parametrization is built and the SSL is applied. Others parameters $\boldsymbol{p}$ than the shape can be added by the user, in this paper the breathing motion is added using one parameter such as $\boldsymbol{p} = b$. This step can be computationally intensive but is easily parallelizable. Computational details and notations can be found in Appendix A, Appendix B and Appendix C. Then, for the *online* stage, the solution is written in a compact canonical tensor format (see Eq. (3)) allowing fast computations. This second step enables the real-time visualization of the solutions through the parametric formulation.

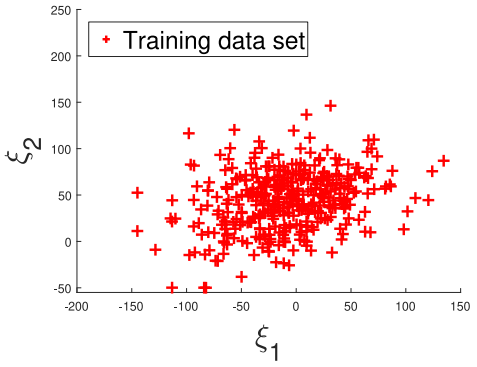
To reach this specific formulation the SSL is used. The method is described in Appendix C. To put it briefly, the work mentioned in Section 1.4 allows to get the FE model from the shape parametrization through a morphing technique. The FE problem defined by Eq. (2) can then be solved for any shape $\tilde{\boldsymbol{X}}(\boldsymbol{\xi})$ (see Eq. (10)). The SSL iteratively computes the solution of this FE problem on well-chosen collocation points, corresponding to specific set of parameters ($\boldsymbol{\xi}$, b). By interpolating these snapshots a prediction of the solution over the whole parameters subspace can then be obtained. At each iteration the interpolation error is reduced and the algorithm stops when it becomes smaller then a given tolerance. The solution is finally written as in Eq. (4) to allow real-time computations. Fig. 5d illustrates the position of the collocation points for the three first increments of the SSL in the shape parameters subspace.

2.5. Model personalization for patient-specific anatomy

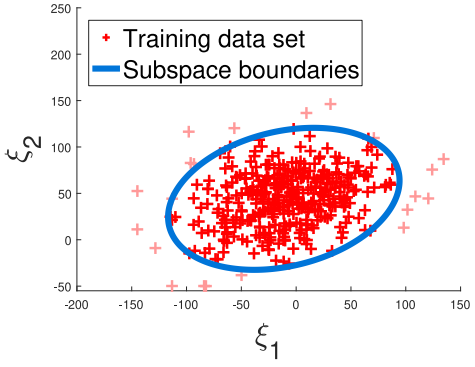
Once the parametric model is built, it can be adapted to a new patient liver anatomy thanks to the shape parametrization. To find the shape parameters associated to a specific anatomy, the idea is to embed the SSMs into the registration technique leading to a reduced complexity formulation. Indeed, with SSM-based registration a $n_p \times n_p$ system needs to be solved on each iteration, which is done on only few seconds. Once the shape parameters are



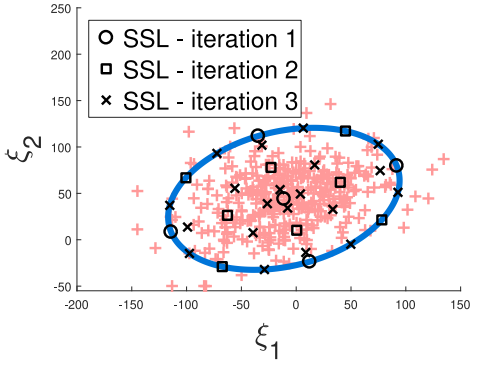
(a) Training data set in initial dimension n, where n is the number of degrees of freedom.



(b) Training data set in dimension $n_p = 2$. A shape is written $\tilde{\boldsymbol{X}}(\xi_1, \xi_2)$ (see Equation 10).



(c) Shape parameters subspace boundaries of the training data set with a 2-hyper-ellipsoid defined by $\mathcal{T}(r, \theta)$ (see Equation 12).



(d) Position of the SSL collocation points in the shape parameters subspace for the three first increments (see Appendix C).

Fig. 5. Illustration of the pROM construction for a 2D case. The steps are applied sequentially. The creation of the statistical boundary shape model and the SSL collocation points are represented in the shape parameters subspace.

estimated, the simulated breathing motion is readily computed as a particularization of Eq. (4) for the new set of shape parameters. Since the parametric solution has been computed offline the amount of online work to obtain this solution is fairly inexpensive and compatible with real-time constraint. Indeed, this particular solution format allows to display at a frequency of 25Hz for visualization up to 1 kHz [37].

It must be noticed that a new shape can lie outside of the boundaries of the parameters subspace. In that case, the nearest shape inside of the parameters subspace boundaries - in terms of Euclidean distance - is considered.

2.6. Validation

For each collocation point of the SSL the rigidity matrix corresponding to the parametrized shape is computed as well as the corresponding displacements of the inner nodes. Considering only the shape, the breathing problem can be seen as adding a shape parameter. The main interest is then to be able to follow the motion of the whole volume when the breathing displacement is applied to a given shape. In a nutshell, without the SSL the following steps need to be applied for each new shape:

1. find the shape parameters,
2. reconstruct the interior of the mesh with the iFEMWRAP,
3. compute the rigidity matrix corresponding to the FE model,
4. solve the FE boundary conditions problem for a given value of the breathing parameter.

With the SSL once the shape parameters are found the solution for any breathing parameter is given by Eq. (4). The reconstruction with the iFEMWRAP, the rigidity matrix computation and the solving are bypassed. This means that the SSL error comes from the "simplification" of these steps. Consequently its error lies in the reconstruction of the interior, either from the iFEMWRAP or from the solving of the FE breathing problem. To assess the accuracy of the SSL "leave-one-out" tests are performed. To this end, the ROM is built using $T - 1$ liver shapes. The last shape is then reconstructed with M modes and the error between the ROM reconstruction and the complete model - i.e. the one built following the 4 steps - is evaluated. Several sources of error are identified:

1. The initial non-rigid registration onto the target shapes. This error is hard to quantify as the correspondence between vertices is not known. That is why the non rigidly registered template is considered as the ground truth afterward in order to be able to compare the vertices position in surface and volume. To estimate the registration accuracy the Volumetric Overlap Error $\text{VOE}(S, S') = 100 \times (1 - |S \cap S'|/|S \cup S'|)$ was evaluated [18], where S and S' represent the compared shapes. An

Table 3
Number of collocation points, i.e. computations, per number of shape parameters n_p as a function of the SSL hierarchical level i. One mode being used for the breathing the total number of parameters is $n_p + 1$. The level i is limited to 3 in order to stay within a reasonable number of computations to do for $n_p = 10$ shape parameters. These values could be increased with more computational ressources. Adding a hierchical level would increase the accuracy of the SSL and adding a shape parameter would improve the shape representation.

i \ n_p	1	2	3	4	5	6	7	8	9	10
1	4	10	14	18	22	26	30	34	38	42
2	4	21	43	73	111	157	211	273	343	421
3	9	58	144	290	512	826	1248	1794	2480	3322
4	20	152	440	1020	2044	3696	6192	9780	14,740	21,384
5	44	384	1264	3304	7392	14,784	27,168	46,728	76,208	118,976

error of 100% means the shapes are completely dissociated, an error of 0% means they perfectly overlap.

2. The projection of the shape on a partial set of principal axis obtained with the SSA. Knowing the solution with the whole set of modes it is possible to compute the error as the distance between corresponding vertices as: $E_p(M) = \frac{1}{n_V} \sum_{\mathbf{s}\in\Omega} ||\mathbf{s} - \mathbf{s}'||$ where Ω represent the whole set of nodes, $\mathbf{s}$ the points of the projected shape and $\mathbf{s}'$ the corresponding points of the ground truth shape. The volume of both shapes is reconstructed with 30 iFEMWRAP iterations.
3. The interpolation done by the SSL approximation of the model. This error can be measured by comparing the results from the iFEMWRAP plus the FE problem on one hand and the SSL on the other hand. This error reads: $E_{SSL}(M) = \frac{1}{n_V} \sum_{\mathbf{s}\in\Omega} ||\mathbf{s} - \mathbf{s}'||$ where Ω represent the whole set of node, $\mathbf{s}$ the points of the SSL rebuilt volumetric mesh and $\mathbf{s}'$ the corresponding points of the FE computed solution done on the iFEMWRAP reconstructed FE model. A distinction is done between $b = 0$, where no FE solution is computed but only the iFEMWRAP, and $b = 1$ where both are.

The maximum global error is the sum of all this independent errors. The tests are done on 20 randomly chosen shapes from the database. The values of E_p and E_{SSL} are then averaged.

In our previous paper [25] the same liver shape database was used to create the SSMs. The quality of the SSA was then evaluated through three metrics: the compactness, the specificity and the generalization. The non-rigid registration method used was found as efficient as the classical Thin Plate Spline registration method.

3. Results

To perform the SSA, the TPS-PR was run with $n_b = 5$ and $\lambda_r = 0.5$ (see Appendix A for parameters description). 35 increments were done for each registration. To create the pROM, the SSL was limited to the third hierarchical level to stay within a reasonable number of collocation points. Table 3 resumes the number of collocation points per level in function of the number of shape parameters. With a MATLAB (The MathWorks, Inc., USA) implementation a computation in a collocation point - i.e. running the iFEMWRAP and solving the FE problem - took between 2 and 30s on an Intel Xeon E5-2680v3 2.5Ghz processor. We used the Parallel Toolbox on 24 cores to drastically reduce the computational time since all computations are independent and can be trivially parallelized.

First, the non-rigid registration VOE is presented in an histogram in Fig. 6. Globally the shapes were well-registered, the error mainly comes from the smoothing of the surfaces by the TPS-PR method which does not particularly deteriorate the shapes representations.

Then, the mean error done by the sole reconstruction is presented in Fig. 7 for $M = 1, \cdots, 100$ shape modes. An example is provided in Fig. 10a. The median value is also represented. It is

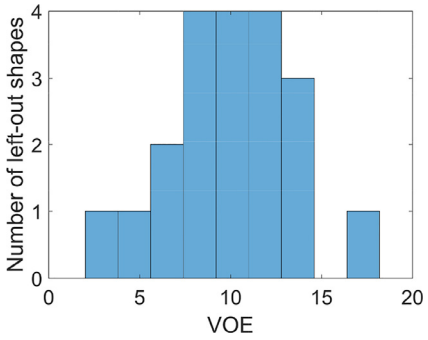


Fig. 6. Histogram of the volumetric overlap error. This error quantifies the goodness of the registration between two closed surfaces by evaluating the percentage of volume not superimposed. A value of 100 means the shapes are dissociated and a value of 0 means they are perfectly superimposed. Here the mean value is around 10% and the standard deviation around 3.5%. Knowing the quality of the shapes from the database these values can be considered as good ones. The error mostly comes from the smoothing of the surfaces.

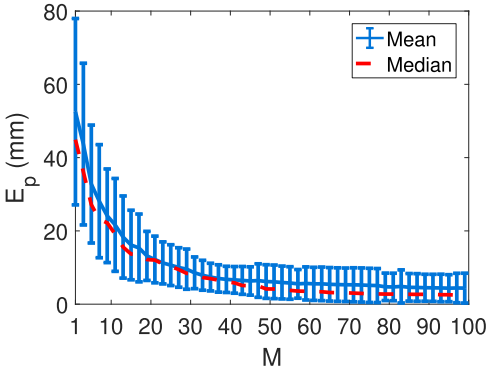


Fig. 7. Error made by the projection on the first M modes. The error bar is one standard deviation. This error shows that a relatively high number of modes (> 50) is required to have a good median reconstruction ($< 4mm$). Some of the database's liver shapes represented in Fig. 1 testify of the shape variability of our database and consequently justify the high number of modes required to have a good representation.

shown that its value is always lower than the mean. This is due to the fact that some of the shapes are very peculiar and consequently require a high number of modes to be represented. That was expected considering that the database contains a large variety of livers, lot of them belonging to ill patients. Because of the large variability of the database both measures of error are quite important. 50 modes are required to get a median error lower than 4mm, this value is given in [20] as the threshold where protocol in radiotherapy could be significantly improved by such tool. Seeking to reach such accuracy can be debated, though. Indeed, there is an initial error in our source data, resulting from image segmentation

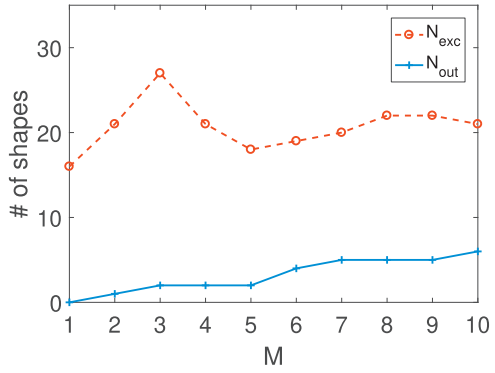


Fig. 8. N_{exc}: number of shapes excluded from the parameters subspace by the point selection. N_{out}: number of sets of parameters taken out of the parameters subspace in the leave-one-out tests. These values depend on the considered number of modes M. N_{exc} is approximately constant and represents around 5% of the number of shapes in the database. On the other hand N_{out} steadily increase with M, meaning that the chance of having a new shape out of the pre-computed solution increases with the dimensionality.

and non-rigid registration. Therefore, although the additional computational effort would improve the model accuracy, it would not increase its precision.

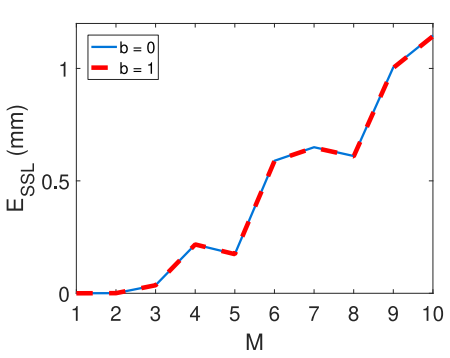
Next, the suitability of the parameters subspace was evaluated. Fig. 8 summarizes how many shapes were removed from the training set to build the n_p-hyper-ellipsoid (see method in Appendix B) and how many reconstructed shapes among the "leave-one-out" tests were outliers, i.e. they were located out of the parameters subspace boundaries. It appears that the number of shapes excluded from the training set varies around 20 and does not seem to depend directly on the number of modes. On the contrary, the number of outliers increases with the number of modes. Such sets of parameters are exceptional and automatically introduce an error as they do not belong to the pre-computed solution and need to be projected onto the n_p-hyper-ellipsoid. For a standard use they should not be considered, that is why a distinction is made afterward between them and the inliers, i.e. the shapes located inside the parameters subspace boundaries.

Eventually, the error between the FE solution and the SSL prediction is shown in Fig. 9a and b. As mentioned before a distinction was made between inliers and outliers. Moreover, the error was computed for $b = 0$ and $b = 1$ to assess the impact of the breathing parametrization on the error. The breathing approximation has a low impact on the error, around $0.1\mu m$. On the contrary, taking shapes out of the parameters subspace introduces a bigger error. With inliers only the maximum error is around 1μm against 1mm when all shapes are taken into account. In both cases the error increases with the number of modes, showing that the convergence is more difficult for higher dimensions. A visual representation of this error is provided in Fig. 10b for $b = 1$.

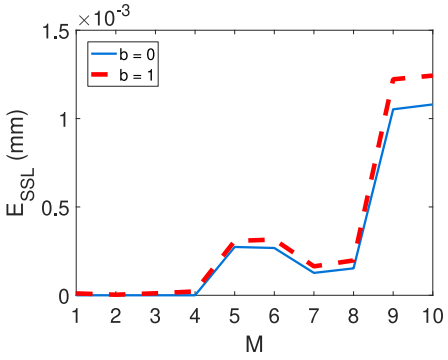
4. Discussion

Globally, the error done in the leave-one-out tests comes from the reconstruction from a limited number of modes. This error could be decreased by using more modes to represent the shape. However, more modes means more computations for the SSL and a more difficult convergence. Here it is difficult to go further than 10 shape modes for the SSL without reaching high computational times. Concerning the SSL error, the main flaw lies in the choice of the parameters subspace. As shown in the previous section the SSL error mainly comes from reconstructed shapes taken out of the parameters subspace, and their number increases with the number of modes. Nonetheless, this error stays within acceptable range and for shapes taken inside the parameters subspace a really high fidelity reconstruction is done. Moreover, a consequent speedup is achieved thanks to MOR while taking into account the same physical equations than the initial FE problem, without making additional assumptions. As noticed in Section 3 this initial model is already computed relatively fast, and with an implementation in a low-level programming language and a more powerful hardware a computation could certainly be done in near real-time ($\sim$1s), which would be sufficient for breathing applications and would question the use of a pROM. As mentioned before the use case presented in this article is a simple example to illustrate the method and more complex situations involving non-linear behaviors should be tested. Anyhow, the pROM approach still has the advantage of requiring simple computational means, which is important to compensate for the lack of advanced hardware in clinical environments.

The first point to tackle to improve the method would be to reduce the number of modes necessary to represent the shape. As the model is based on a large liver shape database coming from medical data, there is an important shape variability. An idea to reduce this variability would be to clusterize the data beforehand. By sorting similar shapes into clusters the variability would be reduced inside each of them [14]. A specific ROM would be then created for each cluster. When a new shape would be introduced, an additional step consisting in finding in which cluster it lies would be done.

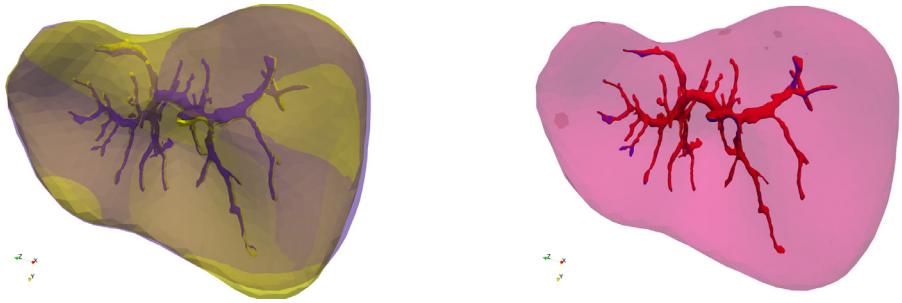


(a) E_{SSL} over the whole left-out training data set.



(b) E_{SSL} over the inliers.

Fig. 9. Representation of the error made by the SSL. The value is averaged over the respective training data sets. Two data sets are compared by distinguishing inliers and outliers. The error is also represented for two extreme values of the breathing parameters b. For inliers only the error is very low ($< 1.5\mu m$). When outliers are taken into account it increases up to $1mm$ for $M = 10$ modes, which is still suitable for AR applications. In both cases the parameter b add an insignificant error ($< 0.1\mu m$).



(a) Ground truth shape (in yellow) versus its projection on the principal axis (in blue).

(b) Projection on the principal axis (in blue) versus the ROM shape (in red).

Fig. 10. Comparison of the shape and the portal vein representations produced by the different steps (the portal vein allows to visually assess the accuracy of the volume reconstruction). The ground truth shape was chosen randomly among the database. The red and blue shapes were reconstructed with $n_p = 10$ shape modes. The breathing parameter was fixed to $b = 1$. The projection on a partial set of principal axis provokes a degradation of the representation as evidenced by the slight shift between the yellow and blue shapes. However, the SSL replicates almost perfectly the projection and the difference is not discernible at naked eye. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The next difficulty to deal with would be the parameters subspace choice. It needs to be a compromise between completeness and efficiency of the SSL while keeping a simple and parametrizable topology. On the contrary to the subspace taken in our previous paper [25], we did not chose a κ%-boundary model [27]. Such subspace considers a normal distribution of the data along each dimension and fix the boundaries separately depending on their standard deviation. Consequently, if one wants to cover 95% of the variability along each dimension, it does not mean that 95% of the whole variation will be covered. In fact the variation covered lies in $([0.95^{n_p}, 0.95] \times 100)$% where n_p is the number of modes - i.e. the dimension. Hence, for an increasing number of modes there is a possible decrease of the variation coverage. That is why the method is not used here and that we use a non-statistical determination of the subspace. The right way to cover κ% of the variation would be to have a multivariate normal distribution of the data. Here, for a number of modes superior to 2 the data do not follow such distribution according to the Henze-Zirkler test [19]. If the data could be clustered with a multivariate gaussian mixture model as proposed before, such choice for the parameters subspace boundaries would be straightforward.

5. Conclusion

A data-based pROM has been developed with the purpose to be integrated into a complete liver Digital Twin. This ROM allows to take into account the patient-specific shape and to model the mechanical deformations caused by the breathing motion. The procedure relies on a SSA in conjunction with the SSL. We used a previously developed registration method, the TPS-PR, to increase its efficiency. The principal novelty of this approach, when compared to literature, is the use of medical data for the shape parametrization of the ROM. A first application to simulate the breathing was done on a simple elastic case, but other applications are envisaged. We showed that, because of the large size of our database, the main limitation is the ability to statistically represent the initial liver shape with a small number of modes. Some leads are suggested to solve this issue, in particular the data clusterization. The whole process works smoothly and provides an efficient way to address the initial matters of patient-specific shape representation within interactive simulation environment, contributing to open the way to the creation of MOR-based tools for patient-specific real-time simulations.

A prototype to visually follow the liver movements during the breathing of any patient is currently being developed at the IRCAD using this approach. This will be the opportunity to validate in vivo the pertinence of this method.

Conflicts of Interest Statement

The authors certify that they have NO affiliations with or involvement in any organization or entity with any financial interest, or non-financial interest in the subject matter or materials discussed in this manuscript.

Acknowledgments

This work was performed by using HPC resources of the Centrale Nantes Supercomputing Centre on the cluster Liger, granted by the High Performance Computing Institute (ICI) (https://ici.ec-nantes.fr/).

This research was financially supported by BPI France through 3D-Surg, a project of the “Programme Investissements d’Avenir – PIA” of the French Government. (http://www.3d-surg.eu/).

Appendix A. Statistical shape analysis

The SSA is done using our previous work [25]. The aim of the SSA is to create the shape parametrization from a training data set of shapes. Two main steps are identified: the registration process and the dimension reduction with the Principal Component Analysis (PCA) [23]. Here, the registration step is unsupervised, meaning there is no *a priori* knowledge on the correspondence of anatomical landmarks between each image. To do this, a well-meshed *template* shape is first selected. Then, all shapes from the database - the *targets* - are rigidly registered to this reference shape. Next, the template undergo a non-rigid registration, called Thin Plate Spline Parametrized Registration (TPS-PR), to fit each one of the targets. The TPS-PR is done by minimizing the sum of two terms, a point maching term I_{match} between the surfaces and the bending energy I_{tps} of a set of control points describing the template through a spline interpolation:

$$I(\boldsymbol{a}) = I_{match}(\boldsymbol{a}) + \lambda I_{tps}(\boldsymbol{a}) \,, \tag{5}$$

where λ is a weighting scalar. The point matching term reads:

$$I_{match}(\boldsymbol{a}) = \frac{1}{2}||\mathbf{A}\boldsymbol{a} - \boldsymbol{d}||^2 \tag{6}$$

and the bending one:

$$I_{tps}(\boldsymbol{a}) = \frac{1}{2}\boldsymbol{a}^T\mathbf{L}\boldsymbol{a} \tag{7}$$

where $\boldsymbol{a}$ is the vector associated to the control points of the spline representation such as $\boldsymbol{u} = \mathbf{A}\boldsymbol{a}$, where $\boldsymbol{u}$ are the displacements of the templates vertices and $\mathbf{A}$ the interpolant operator. The matrix $\mathbf{L}$ is the biharmonic operator of the TPS bending energy. The splines are described through a FE approach using Hermitian elements of the second order. The discretization n_b characterizes the number of control points along each spatial dimension and is user-defined. These points are placed in a regular grid around the template such that they define a bounding box, which global size is fixed to +5% the one of the template. A gradient descent algorithm is used in conjunction with a golden-section search to minimize Eq. (5). As it can be hard to tune the parameter λ, λ_r is introduced as:

$$\lambda = \lambda_r \cdot \frac{\max\limits_{\alpha\in[0,1]} I_{match}(\boldsymbol{a})}{\max\limits_{\alpha\in[0,1]} I_{tps}(\boldsymbol{a})} \qquad \lambda_r \in \mathbb{R}^+ \tag{8}$$

where α is the golden-section search step.

Once the template is non-rigidly registered onto each target, a database of deformations associated to the template bounding box is defined. The PCA is applied to this database in order to extract the n_p principal modes of deformations $\boldsymbol{\phi}_i$. The representation of $\boldsymbol{a}$ in the reduced basis reads as:

$$\tilde{\boldsymbol{a}}(\xi_1, \cdots, \xi_{n_p}) = \sum_{i=1}^{n_p} \xi_i \boldsymbol{\phi}_i \tag{9}$$

where ξ_i are the associated shape parameters. These latter lie within specific ranges defined in Appendix B. Finally, a shape parametrization $\tilde{\boldsymbol{X}}$ can be written as:

$$\tilde{\boldsymbol{X}}(\xi_1, \cdots, \xi_{n_p}) = \bar{\boldsymbol{X}} + \sum_{i=1}^{n_p} \xi_i \mathbf{A}\boldsymbol{\phi}_i \tag{10}$$

where $\bar{\boldsymbol{X}}$ is the mean shape.

Appendix B. Shape parameters subspace boundaries

Once the shape parametrization has been obtained, the subspace where the parameters lie needs to be defined. The definition of this subspace is critical. It must encompass most of the shape variation but must not overpredict the solution. A subspace too large can lead to distorted liver shapes or even tangled meshes. In such cases, the iFEMWRAP method used afterward to rebuild the volume is not insured to converge.

The construction of this subspace is based on the knowledge of the shape parameters associated to the target shapes. Each set of n_p shape parameters defines a shape (Eq. (10)) and a point in dimension n_p (Fig. 5b). First, a point selection is done to exclude exceptional shapes from the training set. The mean Euclidean distance between each point and its k neighbors is computed. Here $k = 30$ is chosen to get a representative averaged distance from a point to its neighbors while keeping clusters if there are some. All values above a threshold are considered as exceptional and the corresponding points are removed. The threshold is empirically fixed to $\bar{d} + 2 \times \mathrm{SD}$, where $\bar{d}$ is the mean distance and SD stands for the standard deviation, in order to keep around 95% of the shapes. Visually, removed points correspond to isolated and peculiar shapes (Fig. 1), consequently removing them should not have an important impact for new shapes reconstruction. Then, the boundaries of the parameters subspace are defined (Fig. 5c). To create the more compact subspace as possible, the minimum volume enclosing ellipsoids method [31] is used. The idea is to generate the n_p-hyper-ellipsoid with the smallest volume in which lie all the training shapes not rejected by the point selection. Each hyper-ellipsoid is defined by its center $c \in \mathbb{R}^{n_p}$, its semi-axes $\boldsymbol{V}_i$ and their associated lengths l_i where $i \in [\![1, n_p]\!]$. As the Euclidean distance between the points depends on the dimension, different training sets will be considered for each n_p-hyper-ellipsoid.

In n_p dimensions the shape parameters ξ_i are now described by a radius parameter $r \in [0, 1]$ and $n_p - 1$ angle parameters θ_i such as $\theta_{n_p-1} \in [0, 2\pi]$ and $\theta_i \in [0, \pi]$ when $i < n_p - 1$. In the reference frame $\mathcal{F} = (c; \boldsymbol{V}_1, \cdots, \boldsymbol{V}_{n_p})$, they are related by:

$$\begin{aligned}
\xi_{1,\mathcal{F}} &= r l_1 \cos(\theta_1) \\
\xi_{k,\mathcal{F}} &= r l_k \prod_{i=1}^{k-1} \sin(\theta_i) \cos(\theta_k) \quad \forall k \in [\![2, n_p - 1]\!] \\
\xi_{n,\mathcal{F}} &= r l_n \prod_{i=1}^{n_p-1} \sin(\theta_i)
\end{aligned} \tag{11}$$

As $(\boldsymbol{V}_1, \cdots, \boldsymbol{V}_{n_p})$ defines an orthonormal basis, the shape parameters are written in the initial reference frame thanks to the bijection:

$$\begin{aligned}
\mathcal{T} : \quad [0,1] \times [0,\pi]^{n_p-2} \times [0, 2\pi] \quad &\to n_p\text{-hyper-ellipsoid} \\
(r, \theta_1, \cdots, \theta_{n_p-1}) \quad &\to \boldsymbol{\xi} = \mathbf{V}\boldsymbol{\xi}_{\mathcal{F}} + c
\end{aligned} \tag{12}$$

where $\mathbf{V} = \left[\boldsymbol{V}_1, \cdots, \boldsymbol{V}_{n_p}\right]$.

Algorithm 1: SSL algorithm.

Inputs : Shape parametrization $\tilde{\boldsymbol{X}}(\boldsymbol{\xi})$ (see Equation 10),
n_p-hyper-ellipsoid associated to $\boldsymbol{\xi}$,
Additional parameters $\boldsymbol{p}$,
Convergence tolerance tol,
Maximum number of iterations i_{max}.
Output: Reduced solution $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{p})$

$i = 1$, $cvg = 0$
$U = []$ `// Create empty matrix`
while $cvg = 0$ and $i \leq i_{max}$ **do**
  Compute values of the collocation points P_j^i ($j \in [\![1, n_c]\!]$)
  for $j = 1$ **to** n_c **do**
    Reconstruct the volumetric mesh of the shape given by $\tilde{\boldsymbol{X}}(\boldsymbol{\xi}^{P_j^i})$ using the iFEMWRAP
    Solve the physical problem associated to the parameters $\boldsymbol{p}^{P_j^i}$ on the volumetric mesh previously computed
    Store the solution - i.e. the snapshot - as $\boldsymbol{u}_j$
  end
  Complete the solutions database $U = [U, \boldsymbol{u}_1, \cdots, \boldsymbol{u}_{n_c}]$
  Get the prediction $\mathcal{P}^i(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{p})$ of the solution over the whole parameters subspace by interpolating the snapshots contained in U
  if $i > 1$ **then**
    $err = 0$
    for $j = 1$ **to** n_c **do**
      $err = \max(err, ||\mathcal{P}^{i-1}(\boldsymbol{x}, \boldsymbol{\xi}^{P_j^i}, \boldsymbol{p}^{P_j^i}) - \boldsymbol{u}_j||)$
    end
    if $err < tol$ **then**
      $cvg = 1$
    end
  end
  $i = i + 1$
end
$\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{p}) = \mathcal{P}^{i-1}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{p})$
Write $\boldsymbol{u}$ in the canonical tensor format (Equation 3)
return $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{p})$

Appendix C. Sparse subspace learning

The SSL [4] is based on the sparse grid approach [6]. It consists in a sparse and compact representation in tensor format of the space of the solutions. When combined with a dimensionality reduction method - e.g. the PGD - the solution can be written in canonical tensor format as in Eq. (3). Such formulation allows to store the data in a compact way and to enable fast computations for real-time applications. A nonlinear greedy algorithm is used to compute a reduced order representation for Eq. (3) that is one with the smallest possible number of terms d.

The method is a constructive and iterative process. We described it here in a framework where the shape is parametrized (see Eq. (10)) and a set of explicit parameters $\boldsymbol{p}$ are used to define the FE model. First, the subspace where the parameters lie is chosen. It will be the n_p-hyper-ellipsoid created in Appendix B for the shape parameters. At each iteration i a point set $P^i = (P^i_1, \cdots, P^i_{n_c})$ is defined following Smolyak's quadrature rule. Note that the value of n_c depends on the iteration i. Each point of this set is a combination of n_p shape parameters plus D other ones such as $P^i_j = (\boldsymbol{\xi}^{P^i_j}, \boldsymbol{p}^{P^i_j})$ $(j \in [\![1, n_c]\!])$. Thanks to the shape parameters $\boldsymbol{\xi}^{P^i_j}$ the iFEMWRAP can be used to get the volumetric meshes for each collocation point. A FE model corresponding to the problem defined by the parameters $\boldsymbol{p}^{P^i_j}$ can then be solved for each P^i_j. The computed solutions are interpolated over the whole parameters subspace to get the prediction $\mathcal{P}^i$. In practice, in the case of n_p-hyper-ellipsoids, the shape parameters are interpolated over the radius and the angle parameters and Eq. (12) is used to get their values in the initial reference frame. Given n_S data points in the shape parameters subspace $(\boldsymbol{\xi}^1, \cdots, \boldsymbol{\xi}^{n_S})$ with all $\boldsymbol{\xi}^k = \mathcal{T}(r^k, \theta^k_1, \cdots, \theta^k_{n_p-1})$ $(k \in [\![1, n_S]\!])$ different and $\left(\boldsymbol{y}^1(\boldsymbol{x}, \boldsymbol{p}), \cdots, \boldsymbol{y}^{n_S}(\boldsymbol{x}, \boldsymbol{p})\right)$ the associated solutions, the radius r is interpolated with the Lagrange polynomials described by:

$$L^k(r) = \prod_{j=1, j\neq k}^{n_S} \frac{r - r^j}{r^k - r^j} \tag{13}$$

and the angle θ_m with the Dirichlet Kernel:

$$\begin{cases} D^k(\theta_m) = \dfrac{sinc\left(\frac{1}{2} n_S(\theta_m - \theta^k_m)\right)}{sinc\left(\frac{1}{2}(\theta_m - \theta^k_m)\right)} \cos\left(\frac{1}{2}(\theta_m - \theta^k_m)\right), & \text{if } n_c \text{ even} \\ D^k(\theta_m) = \dfrac{sinc\left(\frac{1}{2} n_S(\theta_m - \theta^k_m)\right)}{sinc\left(\frac{1}{2}(\theta_m - \theta^k_m)\right)}, & \text{if } n_c \text{ odd} \end{cases} \tag{14}$$

allowing to write the predicted solution on the n_p-hyper-ellipsoid as:

$$\mathcal{P}(\boldsymbol{x}, r, \theta_1, \cdots, \theta_{n_p-1}, \boldsymbol{p}) = \sum_{k=1}^{n_S} \boldsymbol{y}^k(\boldsymbol{x}, \boldsymbol{p}) L^k(r) \left(\prod_{m=1}^{n_p-1} D^k(\theta_m) \right). \tag{15}$$

To insure the interpolation stability the collocation points along the radius are computed using the Gauss-Chebychev-Lobatto points and the angles are equally spaced. The interpolation of parameters $\boldsymbol{p} = (p_1, \ldots, p_D)$ depends on the use case. In this paper, $\boldsymbol{p} = b$ is interpolated using the Lagrange polynomials from Eq. (13). Eventually, the solution is written in a compact format using a low rank approximation as in Eq. (3). As mentioned in Section 1.3 this last step can be done on the fly during the iterations. For the sake of simplicity it is done only once at the end. Algorithm 1 summarizes the different steps and a graphical representation is given in Fig. 4.

The non-intrusivity of the method allows to use external software to run the computations on the collocation points. Moreover, parallelization can be used as collocation points are independent from one another. Table 3 gives an idea of the number of computations required in function of the number of parameters and the hierarchical level. It must be noticed that interpolating with Lagrangian polynomials and Dirichlet Kernel does not involve the same discretization along the concerned dimensions. The first option will require less collocation points, however the parameters subspace may be overestimated. Whatsoever, for a given number of parameters the number of hierarchical level may be guessed by knowing if the expected solution is regular or not in the parameters subspace. As interpolating functions are used to predict the parametric solution, if this latter has a trend similar to a

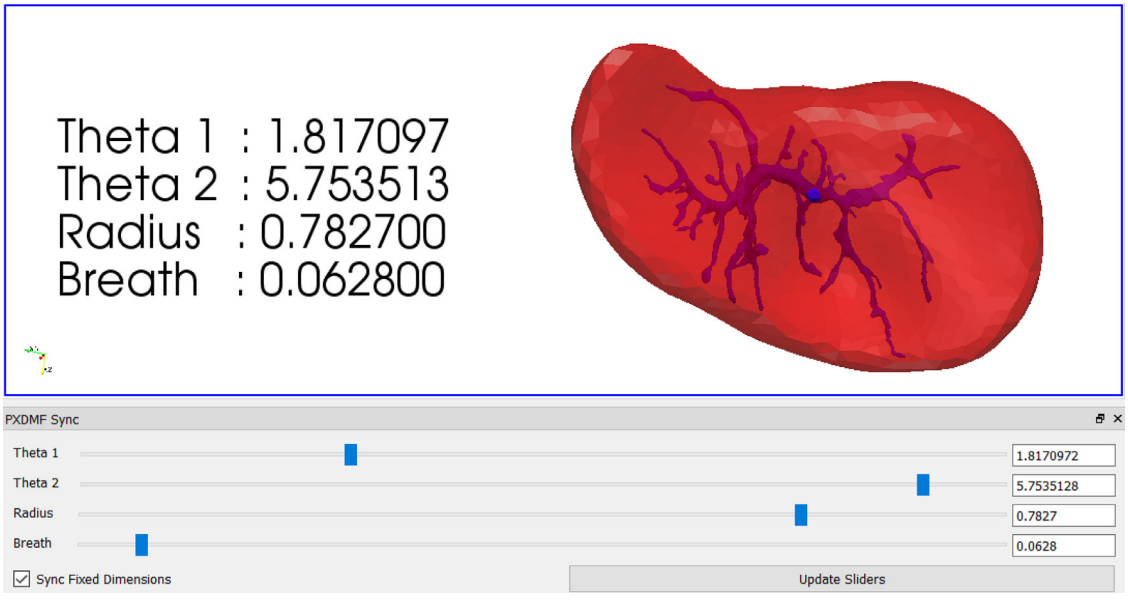


Fig. 11. PXDMF Paraview plugin to visualize separated variables solutions. A solution with three shape parameters plus the breathing one is represented. The sliders at the bottoms allow to visualize in real-time the result for different set of parameters within their predefined boundaries. The radius and angles defined in Appendix B are used instead of the shapes parameters $\boldsymbol{\xi}$, which is equivalent as they are linked by Eq. (12). Here all angles lie in $[0, 2\pi]$ hence some information are redundant.

polynomial, for example, then the Lagrange polynomials will be able to quickly recover the pROM. On the contrary, if the solution is expected to vary quickly in the parameters subspace, then the interpolation will be difficult and a high level of iterations will be required. Consequently, the use of the SSL will depend on the smoothness of the solution variations, the number of parameters, the resources needed for a unique computation and the possibility to use parallelization. Some strategies to avoid the computations of all the collocations points in the SSL can be set up but it will not be detailed in this paper, more insights can be found in [4].

The PXDMF format [3] is used to visualize the parametric solution in real-time via Paraview. Fig. 11 gives an example with 3 shape parameters plus the breathing one.

References

[1] D. Bartz, B. Preim, Visualization and exploration of segmented anatomic structures, in: Biomedical Image Processing, Springer, 2010, pp. 379–401.

[2] S. Bernhardt, S.A. Nicolau, L. Soler, C. Doignon, The status of augmented reality in laparoscopic surgery as of 2016, Med. Image Anal. 37 (2017) 66–90.

[3] F. Bordeu, PXDMF : a file format for separated variables problems, Technical report, Ecole Centrale de Nantes, 2013.

[4] D. Borzacchiello, J.V. Aguado, F. Chinesta, Non-intrusive sparse subspace learning for parametrized problems, Arch. Comput. Methods Eng. (2017) 1–24, doi:10.1007/s11831-017-9241-4.

[5] T.L. Bredbenner, T.D. Eliason, W.L. Francis, J.M. McFarland, A.C. Merkle, D.P. Nicolella, Development and validation of a statistical shape modeling-based finite element model of the cervical spine under low-level multiple direction loading conditions, Front. Bioeng. Biotechnol. 2 (2014) 58.

[6] H.-J. Bungartz, M. Griebel, Sparse grids, Acta Numerica 13 (2004) 147–269.

[7] A. Chatterjee, An introduction to the proper orthogonal decomposition, Curr. Sci. (2000) 808–817.

[8] F. Chinesta, R. Keunings, A. Leygue, The Proper Generalized Decomposition for Advanced Numerical Simulations: A Primer, Springer Science & Business Media, Berlin/Heidelberg, Germany, 2013.

[9] T.F. Cootes, C.J. Taylor, et al., Statistical models of appearance for computer vision, 2004.

[10] E. Cueto, F. Chinesta, Real time simulation for computational surgery: a review, Adv. Model. Simulat. Eng. Sci. (2014), doi:10.1186/2213-7467-1-11.

[11] V. Ferrari, R.M. Viglialoro, P. Nicoli, F. Cutolo, S. Condino, M. Carbone, M. Siesto, M. Ferrari, Augmented reality visualization of deformable tubular structures for surgical simulation, Int. J. Med. Robot.Comput. Assist. Surg. 12 (2) (2016) 231–240.

[12] C. Geuzaine, J.-F. Remacle, Gmsh: a 3-d finite element mesh generator with built-in pre-and post-processing facilities, Int. J. Numer. Method. Eng. 79 (11) (2009) 1309–1331.

[13] D. González, E. Cueto, F. Chinesta, Computational patient avatars for surgery planning, Ann. Biomed. Eng. 44 (1) (2016) 35–45.

[14] A. Gooya, K. Lekadir, I. Castro-Mateos, J.M. Pozo, A.F. Frangi, Mixture of probabilistic principal component analyzers for shapes from point sets, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 891–904.

[15] N. Haouchine, J. Dequidt, I. Peterlik, E. Kerrien, M.-O. Berger, S. Cotin, Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery, in: Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on, IEEE, 2013, pp. 199–208.

[16] N. Haouchine, J. Dequidt, I. Peterlik, E. Kerrien, M.-O. Berger, S. Cotin, Towards an accurate tracking of liver tumors for augmented reality in robotic assisted surgery, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, IEEE, 2014, pp. 4121–4126.

[17] T. Heimann, H.-P. Meinzer, Statistical shape models for 3d medical image segmentation: a review, Med. Image Anal. 13 (4) (2009) 543–563.

[18] T. Heimann, B. Van Ginneken, M.A. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes, et al., Comparison and evaluation of methods for liver segmentation from ct datasets, IEEE Trans. Med. Imag. 28 (8) (2009) 1251–1265.

[19] N. Henze, B. Zirkler, A class of invariant consistent tests for multivariate normality, Commun. Statist.-Theory Method 19 (10) (1990) 3595–3617.

[20] A. Hostettler, D. George, Y. Rémond, S.A. Nicolau, L. Soler, J. Marescaux, Bulk modulus and volume variation measurement of the liver and the kidneys in vivo using abdominal kinetics during free breathing, Comput. Method. Program. Biomed. 100 (2) (2010) 149–157.

[21] A. Hostettler, S. Nicolau, Y. Rémond, J. Marescaux, L. Soler, A real-time predictive simulation of abdominal viscera positions during quiet free breathing, Prog. Biophys. Mol. Biol. 103 (2–3) (2010) 169–184.

[22] G.R. Joldes, A. Wittek, K. Miller, Real-time nonlinear finite element computations on gpu–application to neurosurgical simulation, Comput. Method. Appl. Mech. Eng. 199 (49–52) (2010) 3305–3314.

[23] I.T. Jolliffe, Principal Component Analysis, second ed, Springer, Berlin/Heidelberg, Germany, 2002.

[24] M. Kugler, A. Hostettler, L. Soler, Y. Remond, D. George, A new algorithm for volume mesh refinement on merging geometries: application to liver and vascularisation, J. Comput. Appl. Math. 330 (2018) 429–440.

[25] N. Lauzeral, D. Borzacchiello, M. Kugler, D. George, Y. Rémond, A. Hostettler, F. Chinesta, Shape parametrization of bio-mechanical finite element models based on medical images, Comput. Method. Biomech. Biomed.Eng. (2018), doi:10.1080/21681163.2018.1447400.

[26] D. Lorente, F. Martínez-Martínez, M.J. Rupérez, M. Lago, M. Martínez-Sober, P. Escandell-Montero, J.M. Martínez-Martínez, S. Martínez-Sanchis, A.J. Serrano-López, C. Monserrat, et al., A framework for modelling the biomechanical behaviour of the human liver during breathing in real time using machine learning, Expert. Syst. Appl. 71 (2017) 342–357.

[27] Y.-C. Lu, C.D. Untaroiu, Statistical shape analysis of clavicular cortical bone with applications to the development of mean and boundary shape models, Comput. Method. Program. Biomed. 111 (3) (2013) 613–628.

[28] A. Manzoni, A. Quarteroni, G. Rozza, Shape optimization for viscous flows by reduced basis methods and free-form deformation, Int. J. Numer. Method. Fluids. 70 (5) (2012) 646–670.

[29] F. Martínez-Martínez, M.J. Rupérez, J.D. Martín-Guerrero, C. Monserrat, M. Lago, E. Pareja, S. Brugger, R. López-Andújar, Estimation of the elastic parameters of human liver biomechanical models by means of medical images and evolutionary computation, Comput. Method. Program Biomed. 111 (3) (2013) 537–549.

[30] F. Martínez-Martínez, M. Rupérez-Moreno, M. Martínez-Sober, J. Solves-Llorens, D. Lorente, A. Serrano-López, S. Martínez-Sanchis, C. Monserrat, J. Martín-Guerrero, A finite element-based machine learning approach for modeling the mechanical behavior of the breast tissues under compression in real-time, Comput. Biol. Med. 90 (2017) 116–124.

[31] N. Moshtagh, et al., Minimum volume enclosing ellipsoid, Convex Optim. 111 (2005) 112.

[32] S. Nicolau, L. Soler, D. Mutter, J. Marescaux, Augmented reality in laparoscopic surgical oncology, Surg.Oncol. 20 (3) (2011) 189–201.

[33] D.P. Nicolella, T.L. Bredbenner, Development of a parametric finite element model of the proximal femur using statistical shape and density modelling, Comput. Method. Biomech. Biomed. Eng. 15 (2) (2012) 101–110.

[34] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Real-time deformable models of non-linear tissues by model reduction techniques, Comput. Method. Program. Biomed. 91 (3) (2008) 223–231.

[35] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models, Comput. Method. Program. Biomed. 105 (1) (2012) 1–12.

[36] S. Niroomandi, I. Alfaro, D. Gonzalez, E. Cueto, F. Chinesta, Real-time simulation of surgery by reduced-order modeling and x-fem techniques, Int. J. Numer. Method. Biomed. Eng. 28 (5) (2012) 574–588.

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

[38] A.M. Okamura, Haptic feedback in robot-assisted minimally invasive surgery, Curr. Opin. Urol. 19 (1) (2009) 102.

[39] L. Panait, E. Akkary, R.L. Bell, K.E. Roberts, S.J. Dudrick, A.J. Duffy, The role of haptic feedback in laparoscopic simulation training, J. Surg. Res. 156 (2) (2009) 312–316.

[40] C.J. Paulus, N. Haouchine, D. Cazier, S. Cotin, Surgical augmented reality with topological changes, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 413–420.

[41] A. Quarteroni, A. Manzoni, F. Negri, Reduced basis methods for partial differential equations: an introduction, 92, Springer, 2015.

[42] C. Quesada, D. González, I. Alfaro, E. Cueto, F. Chinesta, Computational vademecums for real-time simulation of surgical cutting in haptic environments, Int. J. Numer. Method. Eng. 108 (10) (2016) 1230–1247.

[43] S.M. Shontz, S.A. Vavasis, Analysis of and workarounds for element reversal for a finite element-based algorithm for warping triangular and tetrahedral meshes, BIT Numer. Math. 50 (4) (2010) 863–884.

[44] S.M. Shontz, S.A. Vavasis, A robust solution procedure for hyperelastic solids with large boundary deformation, Eng. Comput. 28 (2) (2012) 135–147.

[45] O. Van der Meijden, M. Schijven, The value of haptic feedback in conventional and robot-assisted minimal invasive surgery and virtual reality training: a current review, Surg. Endosc. 23 (6) (2009) 1180–1190.

[46] J.-X. Zhao, T. Coupez, E. Decencière, D. Jeulin, D. Cárdenas-Peña, L. Silva, Direct multiphase mesh generation from 3d images using anisotropic mesh adaptation and a redistancing equation, Comput. Methods Appl. Mech. Eng. 309 (2016) 288–306.