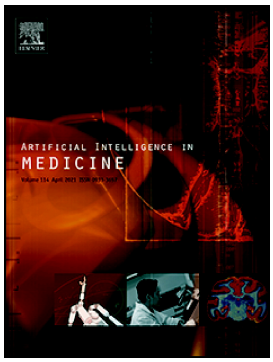
Journal Pre-proof



Neural Network-Based Left Ventricle Geometry Prediction from CMR Images with Application in Biomechanics

Lukasz Romaszko, Agnieszka Borowska, Alan Lazarus, David Dalton, Colin Berry, Xiaoyu Luoa, Dirk Husmeier, Hao Gao

PII: S0933-3657(21)00133-0

DOI: https://doi.org/10.1016/j.artmed.2021.102140

Reference: ARTMED 102140

To appear in: *Artificial Intelligence In Medicine*

Received date: 21 December 2020

Revised date: 10 June 2021

Accepted date: 3 August 2021

Please cite this article as: L. Romaszko, A. Borowska, A. Lazarus, et al., Neural Network-Based Left Ventricle Geometry Prediction from CMR Images with Application in Biomechanics, *Artificial Intelligence In Medicine* (2021), https://doi.org/10.1016/j.artmed.2021.102140

Neural Network-Based Left Ventricle Geometry Prediction from CMR Images with Application in Biomechanics

Lukasz Romaszko[a,1], Agnieszka Borowska[a,1], Alan Lazarus[a], David Dalton[a], Colin Berry[b], Xiaoyu Luoa, Dirk Husmeier[a], Hao Gao[a,*]

[a]*School of Mathematics and Statistics, Univeristy of Glasgow, Glasgow, UK*

[b]*British Heart Foundation Glasgow Cardiovascular Research Centre, Institute of Cardiovascular and Medical Sciences, University of Glasgow, Glasgow, UK*

Abstract

Combining biomechanical modelling of left ventricular (LV) function and dysfunction with cardiac magnetic resonance (CMR) imaging has the potential to improve the prognosis of patient-specific cardiovascular disease risks. Biomechanical studies of LV function in three dimensions usually rely on a computerized representation of the LV geometry based on finite element discretization, which is essential for numerically simulating in vivo cardiac dynamics. Detailed knowledge of the LV geometry is also relevant for various other clinical applications, such as assessing the LV cavity volume and wall thickness. Accurately and automatically reconstructing personalized LV geometries from conventional CMR images with minimal manual intervention is still a challenging task, which is a pre-requisite for any subsequent automated biomechanical analysis. We propose a deep learning-based automatic pipeline for predicting the three-dimensional LV geometry directly from routinely-available CMR cine images, without the need to manually annotate the ventricular wall. Our framework takes advantage of a low-dimensional representation of the high-dimensional LV geometry based on principal component analysis. We analyze how the inference of myocardial passive stiffness is affected by using our automatically generated LV geometries instead of manually generated ones. These insights will inform the development of statistical emulators of LV dynamics to avoid computationally expensive biomechanical simulations. Our proposed framework enables accurate LV geometry reconstruction, outperforming previous approaches by delivering a reconstruction error 50% lower than reported in the literature. We further demonstrate that for a nonlinear cardiac mechanics model, using our reconstructed LV geometries instead of manually extracted ones only

* Corresponding author

Emial address: Gao@glasgow.ac.uk (Hao Gao)

[1] Equal contributor.

moderately affects the inference of passive myocardial stiffness described by an anisotropic hyperelastic constitutive law. The developed methodological framework has the potential to make an important step towards personalized medicine by eliminating the need for time consuming and costly manual operations. In addition, our method automatically maps the CMR scan into a low-dimensional representation of the LV geometry, which constitutes an important stepping stone towards the development of an LV geometry-heterogeneous emulator.

Keywords: 3D reconstruction, MRI, myocardium, convolutional neural network, deep learning

Abbreviations

BO: Bayesian optimization; CMR: cardiac magnetic resonance; CNN: convolutional neural network; GP: Gaussian process; GPR: generalized Procrustes registration; GT: ground-truth; HO: Holzapfel-Ogden; HV: healthy volunteer; LA: long axis; LV: left ventricle; MI: myocardial infarction; MRI: magnetic resonance imaging; MSE: mean squared error; OPR: ordinary Procrustes registration; PCA: principal component analysis; RMSE: root mean squared error; SA: short axis; SBP: systolic blood pressure; SSC: stress-strain curve.

1. Background

Computational studies of left ventricular (LV) mechanics, when integrated with cardiac magnetic resonance (CMR) imaging, can lead to a better understanding of LV dysfunction [1, 2, 3]. For instance, biomechanical parameters that describe LV function provide new insights into the heart's pump function, related e.g. to myocardial stiffness or contractility [4, 5]. Biomechanical studies of LV mechanics typically rely on a discrete representation of the LV geometry [6, 3, 7]. This discretized LV geometry is necessary as the cardiac mechanic equations admit no closed form solution and have to be solved numerically, i.e. using the finite element method [8]. Moreover, such a geometry itself has direct clinical applications as it can be used to derive various cardio-physiological quantities of interest (e.g. the LV cavity volume and local wall thickness), and provide a realistic 3D shape visualization to clinicians. Hence, obtaining an accurate ventricular geometry is an important diagnostic task, especially for personalized health care [9].

Despite the significance of LV geometry reconstruction in clinical applications, so far the literature on how to make this process fast, reliable or automatic has been limited. In general, a typical procedure of LV geometry reconstruction from in vivo CMR data involves the following four steps: 1) segmentation (either manual or automatic) of the LV wall in selected CMR images of a given subject; 2) stacking segmented LV wall boundaries in a selected 3D coordinate system

with necessary motion correction [10]; 3) 3D LV geometry reconstruction either through surface fitting or direct 3D shape generation; 4) generation of a discrete representation of the LV geometry, such as a finite element mesh. As well as requiring specialist knowledge, this procedure is time consuming and prone to human error, which prohibits its wide adoption in the clinic. More recent methods for cardiac geometry reconstruction include manual iterative interventions for reconstruction [11], and warping an idealized ventricular geometry, e.g. an ellipsoid, into patient data [12]. These methods, however, require a separate (manual) segmentation step. In addition, there have been few studies examining the impact on simulations of LV biomechanics when using geometries reconstructed with such techniques in place of manually reconstructed geometries. In particular, reconstructions based on parametric LV geometry representations, like ellipsoids, are likely to result in a systematic bias, which could have potentially severe consequences for subsequent biomechanical analysis.

These difficulties can potentially be addressed with modern machine learning, which has been successfully applied to challenging problems across numerous domains, and its application in medical contexts has the potential to lead to long-lasting advances in healthcare [13]. Image segmentation tasks have attracted particular attention in the medical domain [13, 14, 15, 16, 17]. More generally, convolutional neural networks (CNNs) have proven valuable in various tasks related to cardiac image analysis, such as automated segmentation of CMR scans [14], survival prediction based on sequences of CMR scans [9], or 3D bi-ventricular segmentation from CMR images [18]. Still, comparatively few studies have aimed to learn the ventricular geometry from cardiac images. In the study of Bello et al. [9], for example, the ventricular geometries were not learned directly from CMR images but obtained via a non-rigid registration approach by mapping each patient's data onto a template geometry.

The current paper differs from previous approaches as we go directly from CMR images to the LV geometry by learning its low-dimensional representation. The same idea was initially proposed in [19] where the authors developed a *one-stage* approach with a single CNN predicting the LV geometry from CMR images. In this study we build on this approach by substantially extending the underlying CNN methodological framework. In particular, we propose a *two-stage* method, designating CMR image segmentation and geometry reconstruction to two specialist networks, separately.

The main obstacle for the translation and impact of state-of-the-art cardiac mechanic

models in the clinical setting is the need for patient-specific model calibration and parameter estimation [4]. Traditionally, this is performed using an iterative optimization procedure where a separate numerical solution of the underlying cardiac mechanics equations, obtained using the finite element method, is required at each step of the procedure. The computational costs associated with such an approach make them ill suited to the task of real time decision support [3]. There is currently substantial interest in reducing these computational costs by building a statistical surrogate model or emulator, which would dispense with the need for any finite element simulations. However, an emulator requires a low-dimensional representation of a patient's LV geometry as a functional input, and its manual reconstruction is itself a slow process. For this reason, a method that enables the automatic extraction of such a low-dimensional representation in a fully automated way, directly from CMR images, could pave the way for paradigm-shifting real-time cardiac clinical decision support.

Our main contribution in the present paper is the development of a methodological framework for a fully automated pipeline that provides an automatic extraction of the LV geometry directly from CMR scans. Specifically, we train a convolutional neural network (CNN) to predict the LV geometry by learning its principal component representation directly from CMR scans via automatic pixel labelling. This approach delivers two outputs simultaneously: an LV wall segmentation and a low-dimensional representation of the LV geometry. The motivation for this automatic LV geometry generation is a fully automated procedure for estimation of passive myocardial stiffness. To this end, we investigate the consequences that the automatic generation of LV geometries has on the estimation of passive myocardial stiffness and compare the results with those obtained when using manually generated geometries, as in [20]. Figure 1 presents an overview of the proposed framework.

Figure 1: An overview of the proposed framework: learning 3D LV geometries automatically from CMR images based on a convolutional neural network (CNN). Important applications of the outputted LV meshes are statistical emulators (thanks to the CNN-predicted low dimensional representation of the LV geometry) and parameter inference in cardiac mechanics models.

2. Materials and methods

2.1. Data

In this section we first describe the protocols used to collect the data, including in vivo CMR imaging protocols for both healthy volunteers and patients with acute myocardial infarction. Next, we discuss how the original CMR scans were used to prepare the data for our analysis, consisting of annotated images, LV geometries and corresponding computational finite element meshes. Finally, we summarize the ground truth data used in this study.

2.1.1. Study population and in vivo imaging

The study population consists of 182 subjects; 64 healthy volunteers (HVs) and 118 myocardial infarction (MI) patients. The study was approved by the National Research Ethics Service, and all participants provided written informed consent. All methods, including CMR imaging, were performed in accordance with the relevant guidelines and regulations.

Healthy volunteers with no prior history of cardiovascular disease were enrolled for CMR imaging. A 12-lead electrocardiogram was obtained in all subjects and a normal ECG was an eligibility requirement. Other exclusion criteria included standard contraindication to magnetic resonance such as metallic implants or metallic foreign body. The demographics of all the healthy volunteers can be found in [21]. The MI patients were selected from a larger population of patients with acute ST-elevation MI (STEMI), obtained within a prospective, observational cohort MR-MI study carried out between 14 July 2011 and 22 November 2012, funded by the British Heart Foundation (ClinicalTrials.gov identifier: NCT02072850). Three hundred and forty three patients with acute STEMI were eligible for enrolment in this MR-MI study if they showed signs that they required percutaneous coronary intervention (PCI) due to a history of symptoms consistent with acute MI. Exclusion criteria represented standard contraindication to MR, such as a pacemaker and estimated glomerular filtration rate less than 30 ml/min/1.73m^2. The CMR study of MI patients involved CMR scans at 2.2 ± 1.9 days (the acute state) and 6 months post-MI. Acute STEMI management followed contemporary guidelines. In this study, only the CMR scans at acute state were chosen.

The CMR imaging protocol involved steady-state free precession cine imaging, which was used for LV structure and functional assessment, the short-axis cine stack of the left ventricle from the base to the apex was acquired with 7 mm thick slices and a 3 mm inter slice gap. Typical imaging parameters were: matrix = 180×256, flip angle = 80^o, TR = 3.3 ms, TE = 1.2 ms, bandwidth = 930Hz/pixel, and voxel size = 1.3×1.3×7 mm^3. Standard cine images were also acquired in the LV inflow and outflow tract (LVOT), the horizontal long-axis (HLA), and the

vertical long-axis (VLA) planes. In the STEMI group, typical imaging parameters were: matrix = 192×256, flip angle = 25^o, TE = 3.36 ms, bandwidth = 130 Hz/pixel, echo spacing = 8.7ms and trigger pulse = 2. The voxel size was $1.8\times1.3\times8$ mm^3. The CMR methods and analyses have been previously described in detail in [22].

2.1.2. Non-automated (state-of-the-art) ventricular geometry reconstruction

Below, we describe a non-automated method for ventricular geometry reconstruction, which represents the current state-of-the-art and serves as a benchmark for the automated procedures proposed in the present study. As in the previous study in [4], six short-axis (SA) and three long-axis (LA) cine images were chosen for each subject in order to construct the 3D LV model at early-diastole (when the LV pressure is at its lowest), which is used for further biomechanical analysis. The LV wall boundaries were manually segmented at each imaging plane using in-house Matlab code and the short-axis LV wall boundaries were further aligned to the boundaries in the corresponding HLA, LVOT, VLA images. Details of this manual procedure can be found in [4]. Using these manual segmentations, we obtained the ground truth SA and LA segmentation data by labelling the pixels in the CMR image according to their locations relative to the manually segmented boundaries: 2 for the LV cavity, 1 for the myocardium and 0 for the background.

A prolate spherical coordinate system is used to reconstruct the LV geometry after the manual segmentation and alignment as in [23, 20]. In detail, for a point with Cartesian coordinates (x, y, z), the corresponding spheroidal coordinates (u, v, w) are:

$$\begin{cases} x = \alpha\sinh(w)\cos(u)\cos(v), \\ y = \alpha\sinh(w)\cos(u)\sin(v), \\ z = \alpha\cosh(w)\sin(u), \end{cases} \qquad (1)$$

in which α is a scaling factor, $u\in[-\pi/2, \pi/2)$, $v\in[0, 2\pi)$, and $w\in(0, +\infty)$. After aligning the most-basal plane to the $z = 0$ plane, we have $u = 0$ at the basal plane and $u = -\pi/2$ at the apex. By assuming w to be a cubic B-spline interpolation of u and v, we can fit the endocardial and epicardial surfaces separately using the segmented boundaries from the 6 SA and 3 LA cine images. For details of this surface fitting procedure, the reader is directed to [23]. The LV geometry is the region enclosed by the fitted endocardial and epicardial surfaces, each of which is represented by 2865 quadrilateral patches. In total, 5792 vertices are used for representing one LV geometry. In the Cartesian coordinate system, each vertex has three components, thus the LV

geometry lies in a 17376 dimensional space (denoted "17k").

2.1.3. Data for inference in biomechanical models

Biomechanical meshes. A 3D biomechanical cardiac model requires a volumetric finite element discretization of the LV geometry [3]. To generate it, we first divide the wall thickness between the endocardial and epicardial surfaces into 10 equal divisions, which means for the i^{th} division across the wall, we have:

$$w_i = w_{\text{endo}} + i \cdot \frac{w_{\text{epi}} - w_{\text{endo}}}{10}, \quad (i = 0,...,10).$$

In the same way, equal divisions along the circumferential and longitudinal directions are generated. Finally, a layered hexahedral mesh that is suitable for finite element simulations of LV dynamics is generated. The left panel in Figure 2 presents segmented LV boundaries from an example in vivo CMR cine image and the right panel shows the corresponding computational finite element mesh.

Figure 2: LV geometry reconstruction from a CMR scan of a healthy volunteer. Left: segmented ventricular boundaries (blue: endocardium, red: epicardium) superimposed on a long-axis CMR; right: an example of the reconstructed LV geometry discretized with hexahedron elements.

CMR-derived volumes and strains. In order to infer myocardial passive stiffness from in vivo CMR scans, we measured end-diastolic LV cavity volume and 24 segmental circumferential strains directly from CMR cine images, using procedures similar to [24, 4]. The segments are defined in four short-axial images as specified by the American Heart Association [25].

2.2. Methodology overview

Our primary aim is to develop an automatic method based on convolutional neural networks (CNNs) that reads in CMR cine images and predicts the 3D LV geometry, represented by a cloud of points. This cloud of points can then be used to generate the LV mesh for subsequent cardiac biomechanical modelling. Our approach is based on two cornerstones: using PCA for dimensionality reduction of high-dimensional LV geometries and separating the geometry reconstruction task from CMR image segmentation in a two-stage framework.

Learning a high-dimensional, high-resolution, representation of the LV geometry directly from CMR cine images is a challenging problem, owing to the high-dimensional nature of both the

input and the output domains, which increases the risk of serious overfitting during CNN training. Cutting-edge (deep) machine learning methods are “data hungry" and require dataset sizes much larger than typically available in cardiac studies (see e.g. [16] and the discussion in [26]). For datasets of realistic size, rigorous regularization is needed to prevent overfitting. As we have shown in [27], this loses the non-linear model flexibility that makes (deep) neural networks so powerful in the first place and reduces their predictive performance to that of simple linear predictors.

To address these difficulties, our first cornerstone is to carry out PCA on a large population of LV geometries for dimension reduction, and to map all LV geometries into a low-dimensional space spanned by a few leading principal components. This substantially reduces the output dimension from the order of 17,000 to no more than 10 (the number of leading principal components). This pre-processing procedure, which is hard-coded into our CNN, means that weights on the connections feeding into the output layer of the CNN have been pre-trained and are kept fixed, substantially reducing the network complexity. These fixed, non-adaptable, weights on the edges between the inner bottleneck layer and the output layer of the CNN constitute the PCA basis, where the weights represent the projection into the PCA domain and the number of nodes in the bottleneck layer corresponds to the number of principal components. The effect of this procedure is a novel trade-off between reducing excessive model flexibility that could otherwise cause serious overfitting (by constraining the output weights based on PCA) while keeping enough adaptable degrees of freedom to maintain non-linear model flexibility (all the other weights of the network).

The second cornerstone of our procedure is separating the segmentation and geometry reconstruction tasks. To this end we train two CNNs: a segmentation network (Section 2.4) and a geometry prediction network (Section 2.3). The segmentation network is based on the CNN developed in [14] for segmenting CMR cine images, i.e. assigning distinct labels to pixels (LV wall, LV cavity, and background). Next, the predictions from the segmentation network are used by the geometry prediction network to predict the 3D LV geometry. Below we first describe the latter network as it directly relates to our final object of interest, i.e. the predicted LV geometry. Next, we describe the former network, which provides inputs to the geometry prediction network. Each network is trained independently in a supervised manner on population-wide data and both networks contribute valuable complementary information. The advantage of our two-stage

approach is that we do not need to reconstruct the high-dimensional LV geometries (i.e. 17k components in this study) directly from noisy images because this task is divided between specialist networks. In the Results section (Section 3.3) we show that the proposed two-stage approach achieves a considerable reduction of the LV geometry reconstruction error compared with an approach based on a single-task CNN predicting LV geometries from CMR scans, such as the one in [19]. Figure 3 presents the overview of the proposed approach.

Figure 3: Two-stage framework overview. Given the CMR images, pixel segmentation is predicted using the segmentation network. Next, the geometry prediction network predicts the 3D LV meshes, given the segmentations.

As mentioned previously, our framework delivers two extra outputs on top of the LV geometry. Firstly, we obtain a low-dimensional representation of the high-dimensional LV geometry, which is important in the context of developing statistical emulators. Secondly, for the given dataset, we deliver an automatic segmentation of the LV wall in cine images, in particular in LA cine images, which has not been considered in [14].

2.3. LV geometry prediction network

2.3.1. Preparation: PCA and baselines

Our dataset of LV geometries consists of 182 observations with each 3D LV geometry represented by a 17k-dimensional vector. As discussed above, learning to predict such a high-dimensional vector as a direct function of the CMR image from a relatively small training set size is an ill-posed problem, leading to serious identifiability and overfitting problems. To address this difficulty, we follow [19] and learn a low dimensional representation of the LV geometries using PCA, which substantially reduces the complexity of the LV geometry reconstruction problem. This approach also provides us with a low-dimensional LV geometry encoding, which is paramount for the development of statistical emulators. For evaluation of the reconstruction accuracy we consider two baselines: the results reported in [19] and the predictions obtained using the mean LV geometry. This mean geometry is the LV geometry obtained by taking the mean of coordinates along each of the three dimensions over our LV geometry dataset. More formally, denote the jth GT LV geometry from our dataset, $\mathcal{G}^{*(j)} = \{\mathbf{x}_i^*\}_{i=1}^M$, $j = 1,\mathrm{K}\,,J$, with $J = 182$, where M is the total number of vertices describing the LV geometry, in our case $M = 5792$, and

$\mathbf{x}_i^* = (x_{i,1}^*, x_{i,2}^*, x_{i,3}^*)$ is a vector of 3D Cartesian coordinates of the ith vertex. Then the mean geometry is given as

$$\bar{\mathcal{G}} = \frac{1}{J}\sum_{j=1}^{J}\mathcal{G}^{*(j)}. \qquad (2)$$

2.3.2. Network architecture

The previous work dealing with predicting LV geometries directly from CMR scans [19] considered a CNN consisting of seven layers (five convolutional and two fully connected), taking CMR images (SA and LA views together) as inputs and outputting a four principal component encoding of the LV geometry. Our proposed network architecture (see Figure 4) differs from that of [19] in a number of aspects. The most important and fundamental difference is the splitting of the network into two branches, one each for the aligned SA and LA images, separately. The reason for this split is the inherent difference between LA and SA views. Dividing their processing into two separate CNN branches is therefore a natural "divide-and-conquer" strategy that allows each branch of the CNN to focus on the particular features of their respective views.

To combine the predictions from both branches we further allow each sub-network to output eight PCA coefficients, which we then add via element-wise addition. We have chosen eight PCA components using cross-validation, discussed later in Section 2.3.5. The added PCA coefficient layer is then followed by an eight-unit linear output layer. Our prior experimentation revealed that this approach performs better (in terms of the reconstruction error defined below and convergence speed) than an alternative network with an extra layer[2]. In the convolutional layers we use leaky ReLU ($x = \max(0.2x, x)$) activations.

2.3.3. Inputs and aligning LA images

A significant difference between our geometry prediction network and that in [19] is that we do not use the original CMR images as inputs but the corresponding segmented labelled images. The labelled image is encoded using 3 values: 2 for the LV cavity, 1 for the myocardium, 0 for the background; see the inputs in Figure 4. We then scale the input to the [0,1] range. In addition, we align the LA images so that the long-axis is horizontal. The long-axis is defined as the axis perpendicular to the most basal plane and passing the gravity centre of the LV cavity at that

[2] This additional layer was used for merging the representations from the two branches, each outputting 10 values, followed by a layer of 100 ReLU units and the final linear layer of eight PCA coefficients.

plane (see the lower branch of the CNN in Figure 4). We find that this LA alignment enables a better convergence and more accurate LV geometry reconstruction.

Figure 4: Geometry prediction network: CNN consisting of two identical branches, one for six short axis (SA) images and one for three long axis (LA) images. Each branch consists of five convolutional layers (Conv. 1–5) and a fully connected layer with leaky ReLU activations outputting eight PCA coefficients, which are then added together using the element-wise addition operator. Inputs: LV segmentations encoded 2 for the cavity, 1 for the myocardium and 0 for the background. Output: a left ventricular geometry represented by approx. 17 thousand values (17k).

2.3.4. LV geometry alignment – alternative GT targets

The LV geometries in our original dataset discussed in Section 2.3.3 are aligned such that the gravity centres of the LV cavity in the most-basal SA plane coincide. We will now introduce two additional datasets, derived from the original basal-aligned dataset that will serve as alternative GT targets for the geometry prediction network (see also Figure 7). The basic idea is to align the LV geometries in a way that will make it easier for the geometry prediction network (see Figure 4) to learn the target LV geometries. In particular, we consider its two variants: ordinary Procrustes registration (OPR) and generalized Procrustes registration (GPR).

OPR uses the operations of rotation, scaling and translation to best match a discretized shape $\mathbf{X}$ to another shape $\mathbf{Y}$. This is expressed as

$$\min_{\beta,\boldsymbol{\Gamma},\boldsymbol{\gamma}} \left\| \mathbf{Y} - \beta \mathbf{X}\boldsymbol{\Gamma} - 1_J \boldsymbol{\gamma}^T \right\|^2, \qquad (3)$$

where β and $\boldsymbol{\gamma}$ are the scaling and translation parameters respectively, and $\boldsymbol{\Gamma}$ is the rotation matrix. $\mathbf{X}$ and $\mathbf{Y}$ are matrices of equal dimension, where each row gives the position vectors of each point in the discretized geometry. In addition, both matrices are assumed to have centroid zero.

GPR expands on OPR to align a set of shapes $\mathbf{X}_1, ..., \mathbf{X}_N$ to a reference geometry $\bar{\mathbf{X}}$ as closely as possible. This is done by iteratively minimizing

$$\sum_{i=1}^{N} \left\| \bar{\mathbf{X}} - \beta_i \mathbf{X}_i \boldsymbol{\Gamma}_i - 1_J \boldsymbol{\gamma}_i^T \right\|^2, \qquad (4)$$

with respect to the transformation parameters $\{(\beta_i, \boldsymbol{\gamma}_i, \boldsymbol{\Gamma}_i)_{i=1}^N\}$, where at each iteration the

reference shape $\bar{\mathbf{X}}$ is the mean of the transformed shapes from the *previous* iteration. Again, the shapes are assumed to have centroid zero. This procedure is then iterated until convergence.

Our first alternative dataset alignment – the result of which we call "centered-GT" – is based on GPR where only translations of the LV geometries are considered. Under the standard Euclidean metric, the optimal translation corresponds to simply aligning each LV geometry to have a common centroid, which we set to the origin. The second alternative alignment is constructed using GPR where only translation and rotation operations are used, which we refer to as "rotated-GT". We do not unify the scaling of the LV geometries because we want to predict the size of each LV geometry explicitly. Note that the rotations applied to the LV geometries by GPR were small as all the LV geometries in the original dataset share a common orientation. Any further rotation is to account for intrinsic shape variation in each specific LV geometry.

2.3.5. Training

First, we train the geometry prediction network using the ground-truth (GT) segmentations as inputs, which we have described in Subsection 2.1.2. Later on, in Section 3.3, we will discuss the results of the two-stage framework in which the geometry prediction network is trained using the segmentations predicted by the segmentation network discussed in Section 2.4.

In the training we minimize the LV geometry reconstruction error taken as the mean squared error (MSE) between the GT LV geometry and its reconstructions. See Section 6.1 in [28] for a justification of the MSE as an objective function. Formally, using the notation introduced in Section 2.3.1, let $\mathcal{G}^{*(j)}$ denote the j th GT LV geometry from our dataset, $j = 1,\mathrm{K}\,, J$. Furthermore, denote the corresponding predicted geometry $\mathcal{G}^{(j)=}\{\mathbf{x}_i^{(j)}\}_{i=1}^{M}$. We then calculate the MSE between $\mathcal{G}^{*(j)}$ and $\mathcal{G}^{(j)}$ as:

$$\mathrm{MSE}^{(j)}:=\mathrm{MSE}(\mathcal{G}^{*(j)},\mathcal{G}^{(j)}) = \frac{1}{3M}\sum_{i=1}^{M}\sum_{k=1}^{3}(x_{i,k}^{*(j)} - x_{i,k}^{(j)})^2. \tag{5}$$

Finally, we consider the average MSE over the whole dateset of J LV geometries calculated as:

$$\mathrm{MSE}= \frac{1}{J}\sum_{j=1}^{J}\mathrm{MSE}^{(j)}. \tag{6}$$

We use the average MSE (6) to train the geometry prediction network and to assess the reconstruction accuracy for the different datasets (see Figure 7 for more details).

To prevent overfitting we use 14 fold cross-validation, i.e. we divide our dataset of 182

subjects into 14 folds, with 13 subjects each, with all 182 subjects (HV and MI) randomly shuffled. We then train the network on 12 folds and use one of the remaining two folds for cross-validation (to be discussed below) and the other one for testing. Importantly, PCA is performed only on the training data, without the test or validation data. We then set the number of PCA components to the number that minimizes the cross-validation error (leading to an optimal number of 8 components).

As the CMR images usually have different pixel spacing, we first unify them by applying a common pixel spacing so that all CMR images and corresponding labelled images are expressed in the same resolution. We then crop 64×64 pixel patches from the unified labelled images so that the LV cavity centre is in the centre of the crop. We then take 60×60 random crops from the 64×64 image patches for training, and the central crop of the same size for validation and testing. Taking random crops for training corresponds to a crop noise of $[\pm2\times\pm2]$ pixels. Random cropping is a standard data augmentation technique in computer vision, aimed at preventing overfitting, see [29] for details.

The CNN training makes use of two regularization parameters (often referred to as "hyperparameters" in the deep learning literature): the learning rate and the L2 regularization strength. In addition to controlling the convergence rate, the learning rate also provides regularization given a fixed number of training epochs via early stopping. The L2 regularization strength prevents overfitting by keeping the weights of the CNN low. This is done by adding the squared L2 norm of the total weight vector, multiplied by the L2 regularization strength, to the standard MSE loss function. In some layers (see Figure 4 for details) we also apply another regularization technique called *dropout* [30]. Dropout involves randomly switching a subset of the networks weights to zero at a given rate.

The values of the CNN regularization parameters and the dropout rate are the same for each cross-validation fold, with the L2 regularization parameter set equal to 0.001, the learning rate to 0.0004 and the dropout rate to 0.01. We select these values manually, based on the cross-validated CNN performance after 300 epochs (following [19]). We implement the geometry prediction network using Python and TensorFlow. Training of a single network takes about two minutes on a NVIDIA Quadro P4000 GPU for each cross-validation partition.

2.4. Segmentation network

To implement our segmentation network, we use the freely-available network architecture

from [14], which we display in Figure 5. This architecture is similar to the U-net architecture from [31] but without intermediate up-convolutional layers.

Figure 5: Segmentation Network (adapted from [14]): the network learns image features from fine to coarse scales through a series of convolutions (Conv. 1–16), then upsamples (with transposed convolutions, transp., and concatenates (concat.) multi-scale features, to finally predict the pixel-wise image segmentation (one of the three label classes: background, myocardium, cavity). The illustrated example image is of the LA view of the LV (not considered in [14]), but the SA view is processed in the same manner.

Training. We need to train the network from [14] on our own dataset, individually for SA and LA images, since the network originally trained in [14] does not segment the LV wall in LA images. Moreover, we also found it did not work properly for the SA images from our dataset[3]. The general approach to training our segmentation network is similar to the one outlined for our geometry prediction network in Section 2.3.5. There are, however, some changes to the training procedure that we will now outline in more detail. Firstly, we resize the CMR images as the inputs to the network with a size of 160×160 pixels. As in [14] we use the mean cross entropy as the loss function for training; see Section 6.9 of [28] for a discussion of why this is the optimal loss function for discrete response variables. The cross entropy is computed between the GT annotations and the probabilistic label predicted by the network, training the network for 2000 iterations, using a batch size of four. The latter is due to the memory limit available on our GPU as the network uses approximately 1GB of GPU memory per image. Similar to the geometry prediction network, we perform 14-fold cross-validation. This means that each training split uses 12 folds with 13 subjects each, with each subject consisting of 6 SA and 3 LA images, resulting in 936 SA images (12×13×6) and 468 LA images (12×13×3) for training. We explore various data augmentation techniques when training the segmentation network, such as the addition of noise to the image scale, as well as image rotations and shifts, but none yields performance improvements.
Evaluation. For the segmentation task we use the Dice score, separately for the LV wall and the LV cavity, as the evaluation measure. The Dice score is defined as $2|P \cap G|/(|P|+|G|)$, where

[3] It often worked reasonably well for the basal and mid-ventricle SA images of healthy volunteers, but not for SA images close to the apex, and it performed poorly for MI patients.

P and G are the predicted and ground-truth pixel sets, respectively. The Dice metric is also used in [14]. It is similar to another popular criterion called intersection over union (IoU), defined as $|P \cap G| / |P \cup G|$. However, Dice focuses more on the average prediction rather than on the worst-case prediction, as IoU does.

2.5. Passive myocardial stiffness inference using biomechanical models

Having outlined the automatic LV geometry generation, we will now discuss how this can be combined with a biomechanical model of the left ventricle to approximate the behaviour of the left ventricle in diastole and how we plan to asses the accuracy of the LV geometry representation in this context.

The biomechanical model, which is visualized in Figure 6, can simulate the diastolic filling process from early to end of diastole. Important for such a model is the strain energy function, describing the build up of energy in the tissue as it is deformed. In our forward model this is provided by the Holzapfel-Ogden (HO) law [32], a detailed discussion of which can be found in the literature [32]. For the work presented here, it is only important to know that this function contains eight constitutive parameters $a, b, a_f, b_f, a_s, b_s, a_{fs}, b_{fs}$ that, when accurately inferred, allow us to describe the passive properties of the cardiac tissue.

The parameters of the HO law cannot be measured in vivo and must be inferred non-invasively from indirect measurements, which typically involves some expensive iterative optimization procedure. This idea was used in a specific multi-step procedure for inferring myocardial properties using the HO law in [24]. Since that study inferring HO law parameters has been considered in a series of subsequent studies [4, 33, 34].

This method involves matching the model-predicted LV volume (V) and 24 circumferential strains ε_i, $i = 1,\text{K}\,,24$, to the measured ones, extracted from in vivo CMR scans (see Section 2.1.3). Denoting the volume and strains by V^* and ε_i^*, $i = 1,\text{K}\,,24$, considered the following loss function [24]:

$$loss = \sum_{i=1,\text{K}\,,24} (\varepsilon_i - \varepsilon_i^*)^2 + \frac{(V - V^*)^2}{V^*}, \qquad (7)$$

which can be optimized to find a set of best fitting parameters. Details on the forward biomechanical LV model and CMR derived strains and volumes can be found in [24]. The LV model is simulated for diastolic filling with a population-based end-diastolic pressure, taken to be

8 mmHg. This will allow us to infer myocardial passive properties as in [24].

Figure 6: Schematic illustration of simulating LV dynamics in diastole. Left: inputs to the model, right: outputs from the model. Given a representation of the left ventricle (LV) in the form of a finite element mesh at a given reference time point t_1 of the pump cycle, blood pressure as a boundary condition and the constitutive parameters of the cardiac mechanic model, the shape of the left ventricle at a later time point t_2 can be predicted. In our study, t_1 is early diastole, and t_2 is end-diastole. The LV at early diastole, t_1, has to be extracted from CMR scans (indicated by the arrow on the very left), and automating this process is the main purpose of our work. We assess the performance of our method not only in terms of the LV geometry reconstruction itself, but also in terms of the impact that a perturbation of the LV geometry has on cardio-mechanic processes. To infer the cardio-mechanic parameters, we compare the LV geometry at t_2, as predicted by the cadio-mechanic model, with the LV extracted from a CMR scan at the same time point (late diastole), quantify the mismatch, and use Bayesian optimization to find the cardio-mechanic parameters that minimize the corresponding objective function. To allow for multi-modality of the objective function and potential weak identifiability of some of the parameters, we use the cardio-mechanic parameters thus inferred to compute stretch-stress curves along different directions in the myocardium, and then assess the accuracy of our method in stretch-stress curve space.

The eight parameters of the HO law are challenging to infer because they are highly correlated and weakly identifiable, see [24]. Therefore, we follow the approach proposed in [33] and reparametrize the original eight-dimensional inference problem [5] using a four-dimensional vector $\boldsymbol{\theta}$ as follows

$$\begin{aligned} a &= \theta_1 a_0, \quad b = \theta_1 b_0, \quad a_{\mathrm{f}} = \theta_2 a_{\mathrm{f0}}, \quad a_{\mathrm{s}} = \theta_2 a_{\mathrm{s0}}, \\ b_{\mathrm{f}} &= \theta_3 b_{\mathrm{f0}}, \quad b_{\mathrm{s}} = \theta_3 b_{\mathrm{s0}}, \quad a_{\mathrm{fs}} = \theta_4 a_{\mathrm{fs0}}, \quad b_{\mathrm{fs}} = \theta_4 b_{\mathrm{fs0}}, \end{aligned} \tag{8}$$

where the nought subscripts indicate the reference values, which can be taken from published studies [8, 5]. Details on the reparameterization can be found in [33, 5].

2.5.1. Bayesian optimization

A disadvantage of the approach in [24] is its high computational complexity and the fact

that it is not guaranteed to converge to the global minimum of the objective function (7). To circumvent these problems we use Bayesian optimization (BO), which is a global optimization method based on sequential training of a statistical approximation to the unknown target function. BO is particularly suited for problems in which a single evaluation of the objective function is time consuming. We refer to [35] for a review of the BO methodology.

As in [36], we consider two versions of BO, based on "target surrogates" and "partial error surrogates". In the former approach the objective function (7) is approximated as a whole using a single Gaussian process (GP) regression, while in the latter approach each error term in (7) is approximated separately, using a separate GP regression. We refer to [37] for a detailed treatment of GP methods. Note that in this paper, our interest is not in comparing different variants of BO, but to verify how robust parameter inference is by considering two independent runs of BO for basic uncertainty quantification.

We initialize BO using an initial design based on the Latin hypercube, following the standard recommendation [38] to use $10\times D$ points, where D is the dimensionality of the problem (in our case the number of parameters to be inferred).

2.5.2. Evaluation methods

We are interested in comparing the parameter inference performance when using the LV geometries obtained automatically from our proposed CNN-approach with the results obtained using the manually segmented geometries (see Sections 2.1.2) and the baseline of the mean geometry (defined in Eq. (2)). To make this comparison, we make use of three separate evaluation methods, which we outline below.

BO is designed to optimize a given objective function, thus the best objective function value obtained by the BO optimisation routine will serve as our first evaluation method. Here we are interested in the minimum of the mismatch function (7) so we prefer approaches leading to lower *loss* values. A low value of the objective function indicates that the underlying parameters have led to outputs (LV volume and circumferential strains) closely agreeing with the CMR-derived measurements described in Section 2.1.3 (real or synthetic).

For synthetic data, generated with known GT parameters, we can assess the inference accuracy in the parameter space by calculating the weighted L2 distance between the GT and inferred values. In particular, we are interested in the relative root mean squared error (RMSE), calculated as

$$\text{rRMSE} = \left(\sum_{i=1}^{D} \left(\frac{\hat{\theta}_i - \theta_i^*}{\theta_i^*} \right)^2 \right)^{1/2}, \qquad (9)$$

where θ_i^* denotes the GT value of θ_i, $\hat{\theta}_i$ is the estimate of θ_i and D is the dimensionality of the problem, in our case $D = 4$. Scaling each error term by the corresponding GT value introduces some robustness against inflation by large parameter values.

The problem with evaluating in the parameter space is that the parameters of the HO model are not guaranteed to be uniquely identifiable from the available experimental data, as discussed e.g. in [24]. Furthermore, the passive behaviour of the myocardium is highly nonlinear with respect to its stretch. For that reason, differences in the biomechanical parameter space may not be informative about the difference in myocardial stiffness and can even be misleading. On the contrary, the stretch-stress space derived from the HO model under a prescribed stretching mode (i.e. uniaxial) is directly related to myocardial stiffness at different stretch levels, and is often employed in ex vivo stretching experiments [39] or material parameter inference studies [24, 20]. To derive the stretch-stress space for a specified direction, we virtually stretch a myocardial strip uniaxially at one end while keeping the other end fixed. The corresponding stress in that strip is then calculated using the constitutive law by assuming homogeneous stretch occurring in the entire strip [39, 24, 32]. Note that stretching a myocardial strip is similar to stretching a nonlinear rubber band. In this way, two virtual myocardial strips are considered here. One strip is along the myofibre direction with stretch $\lambda_f = l / L$, where l and L are the current and reference lengths of the same myocardial strip, respectively. The other one is along the sheet direction with $\lambda_s = l / L$. The interested reader may refer to [40] for some examples of uniaxial stretch-stress derivation using a similar HO model. These stretch-stress curves allow us to gain insight into the underlying myocardial stiffness values, because in general, higher stress values for a given stretch level indicate higher stiffness. Therefore, as an additional evaluation metric we use stretch-stress curves along those two principal directions (myofibre and sheet directions) according to a layered myofibre structure approximation.

3. Results

In this section, we firstly present the individual results of the segmentation and geometry prediction networks, where each network is applied separately. Then, we present the results of the two-stage approach for LV mesh reconstruction, where both networks are applied together in

sequence. Finally, we quantify the utility of these reconstructed meshes for parameter inference in a bio-mechanical model of the LV. Figure 7 illustrates the evaluation methods that will be used and the datasets of interest in each case.

Figure 7: An overview of the evaluation methods and datasets used for CMR image segmentation, geometry reconstruction and parameter inference respectively. For parameter inference "$\times 2$" refers to repetitions of Bayesian optimization with two algorithms for basic uncertainty quantification. Dashed arrows indicate how the datasets are related. For assessing the accuracy of parameter inference, we used synthetic data. These were generated from simulations using the material parameters estimated from the real data. We represent this graphically in the figure with a dashed horizontal arrow, which indicates that the generation of the synthetic data has been informed by the real data.

3.1. Geometry prediction network

The geometry prediction network, which is visualized in Figure 4, requires segmented CMR scans as inputs. When the segmented scans are obtained automatically using the segmentation network, they will inevitably be subject to a degree of error, which in turn will affect the accuracy of the predicted geometry. For this reason, we first present the results of the geometry prediction network where the manually obtained, GT segmentations are provided as input. These results provide a lower bound on the error that can be achieved using our fully automated, two-stage approach. In addition, this step allows us to determine the optimal number of PCA coefficients to use in the output layer of the segmentation network. Using more coefficients increases the flexibility of the model, allowing for more complex LV geometry features to be accounted for. However, using more coefficients also increases the risk of the network overfitting to the training data. We find that eight PCA cofficients optimially balances the trade-off between these considerations. (Note that this differs from the four coefficients used in [19]). With eight coefficients, the geometry prediction network achieves a MSE of 0.03 when the GT segmentations are provided as inputs, which is 40% less than the MSE of 0.049 reported in [19].

3.2. Segmentation network

As discussed in Section 2.4, we retrain the network from [14] to segment SA and LA images separately. The segmentation results for the two image types respectively are discussed in

turn below. The results are evaluated using the DICE score metric on both the wall and cavity of the LV, as is illustrated in the top panel of Figure 7.

3.2.1. Short axis images

Table 1 summarizes the SA segmentation results for the entire dataset. When training from scratch with randomly initialized weights, we obtain average Dice scores of 80.5% for the LV wall, and 90.9% for the cavity. When training with weights initialized using the pre-trained values from the network published in [14], improved Dice scores of 85.0% and 92.5% respectively are obtained. These improved scores are comparable with the scores of 88% for wall and 94% for cavity found in [14], which were based on a much larger dataset of approximately 5000 subjects. Note that for each of the approaches displayed in Table 1, the Dice score of the cavity segmentation is higher than the wall segmentation score. This is unsurprising, since the higher contrast of the cavity makes this easier to segment than the LV wall, which is sometimes indistinguishable from surrounding tissues.

Figure 8 displays a comparison of the GT segmentations of a HV with the predictions from the segmentation network. We see that the predictions of the network get less accurate near the apex of the LV. However, it is worth noting that due to the lack of a clear boundary in the CMR scan, the manual segmentation also becomes more erroneous in this region.

Table 1: SA segmentation network results: Dice scores for different network initializations and those from [14] for comparison.

Case	Wall	Cavity
Random weights	80.5%	90.9%
Weights from [14]	85.0%	92.5%
Results from [14]	88.0%	94.0%

Figure 8: SA segmentation network results. Clockwise from top left: typical segmentation results as we move from basal plane to apex for one healthy volunteer. The red line shows the manual segmentation results while the shaded regions show the segmentation network prediction. As we move towards the apex, segmentation becomes more difficult both for manual and automatic methods.

3.2.2. Long axis images

We train the LA Segmentation Network using all three LA views together, again initializing with the pre-trained weights from [14]. Table 2 displays the mean Dice scores attained by the network, broken down for each of the LA views. We can see that in particular, wall segmentation scores were lower for the LA images when compared to the SA images. Figure 9 displays a comparison between the GT and automatically predicted LA segmentation for one of the subjects. The obtained mean Dice score is 86.60%, which is slightly lower than that for SA (88.75%). However, despite this slight deterioration, we can see that our segmentation network still delivers a reasonably accurate segmentation of LV wall in the long-axis view.

Table 2: LA Segmentation network results: Dice scores for 3 different long-axis views.

View	Wall	Cavity
2 chamber	87.6%	93.4%
4 chamber	81.7%	93.3%
1 chamber	76.3%	92.3%
Mean	80.2%	93.0 %

Figure 9: LA segmentation network results. Predictions for all three LA views for a single subject. Top left provides the four chamber view, top right shows one chamber view and bottom is a two chamber view.

3.3. The two-stage framework

We now report the results obtained by our two-stage reconstruction framework, whereby the CMR images are first segmented using the segmentaion network, before these segmentations are passed to the geometry recontruction network, which predicts the LV geometry. The middle panel in Figure 7 outlines how the reconstruction results are evaluated using MSE between the GT and predicted geometries respectively. Note that we found the MSE scores to be very stable - running the CNN several times with different random number generator seeds varied the results usually only in the fourth digit. For this reason, we report the MSE scores up to the third digit.

When trained on the automatically segmented SA and LA images from the segmentation network, the geometry prediction network results in an MSE of 0.033 relative to the GT geometries. This is only 10% higher than the lower bound of 0.03 described in Section 3.1, when the network was trained on the GT, manually segmented SA and LA images. As expected, the predictions of the network deteriorated when either only the SA or only the LA images were provided as input: for the SA-only network, the MSE loss was 0.038, while for the LA-only network, the MSE was 0.044. These results are summarized in the top half of Table 3.

The bottom half of Table 3 summarizes the results of the two-stage approach where the LV geometries are aligned using the two Procrustes techniques described in Section 2.3.4. The results show that carrying out these alignments results in further reconstruction improvements: for the centred dataset, the MSE decreases to 0.026, and there is a further slight gain for the rotated dataset, where the MSE was 0.025. Note however that by pre-aligning the LV geometries, the performance of the baseline mean-geometry prediction also improves: the MSE declines from 0.074 for the original data set to 0.058 and 0.057 for the centered and rotated datasets respectively. The cause of this improvement is the fact that by aligning the datasets in this manner we reduce the variance between the individual geometries, and hence reduce the MSE of the mean-geometry prediction.

Table 3: The two-stage approach results: MSE (in mm^2). The top part of the table reports the results for the original dataset of LV geometries, and the bottom part presents the results for the two alternative alignments of the dataset of LV geometries described in Section 2.3.4.

Model	MSE (mm^2)
Dataset of original LV geometries	
Mean geometry	0.074
CNN baseline from [19]	0.049
GT-segmentation CNN lower bound	0.030
Segmentation-based CNN (SA-only)	0.038
Segmentation-based CNN (LA-only)	0.044
Segmentation-based CNN	0.033
Dataset of aligned LV geometries	
Mean geometry – centered-GT	0.058

Mean geometry – rotated-GT	0.057
Segmentation-based CNN (centered-GT)	0.026
Segmentation-based CNN (rotated-GT)	**0.025**

Table 4 shows the results of the two-stage reconstruction approach on the test set for different number of PCA components. The results show that reconstruction performance is robust, even as we vary the number of components over a substantial range from 4 to 10. Note that in our main simulations, we used eight components consistently, based on preliminary explorations described in Section 3.1. That is, we did *not* select the number of PCA components based on the test set results from Table 4, as doing so would introduce selection bias.

Table 4: MSEs (in mm^2) for the two-stage approach for different numbers of PCA components. The first four components are the most important and were used in [19]. For the segmentation input we used eight components. The minimum is obtained for eight and seven components for the original and centered/rotated data, respectively.

Number PCA components	Original dataset	Centered dataset	Rotated dataset
4	0.035	0.026	0.025
5	0.035	0.027	0.025
6	0.034	0.026	0.025
7	0.034	0.025	0.024
8	0.033	0.026	0.025
9	0.033	0.026	0.025
10	0.033	0.025	0.024

Figure 10 presents examples of automatically generated LV geometry reconstructions from our two-stage approach, against the corresponding GT geometries. The examples presented are typical cases, for which the reconstruction errors are approximately equal to the median reconstruction error. The reconstructions are accurate, particularly with respect to the general size and height of the LV, but some small misalignments occur near the edges.

Figure 10: The two-stage approach example reconstructions. Depicted are typical cases –

reconstructions that obtained approximately the median reconstruction error. Blue shape: reconstruction, grey wireframe: GT. The reconstructions are accurate, LV sizes and heights are similar, there are small misalignments at the edges.

As an additional experiment, we calculate the cavity volumes of the LV geometries automatically generated by our two-stage method, and compare with the volumes of the GT geometries. As shown in Figure 7, we perform this experiment only for the data set of rotated LV geometries, on which our automated method incurred the lowest reconstruction MSE. On this dataset, our method achieves an average RMSE of 12 ml in volume. This is lower than the RMSE of 26 ml obtained using the mean LV mesh baseline, and is comparable with values from the literature. For instance, the network developed in [41], which was trained using a volume-based objective on a larger dataset, achieved an RMSE of 10 ml.

3.4. Parameter inference

Having evaluated the LV geometry reconstruction in geometry space, we now investigate the effect this on the inferred mechanical properties as discussed in Section 2.5. We make use of four LV geometries corresponding to randomly selected HVs, which we label HV A, HV B, HV C and HV D. We consider three ways to represent each LV geometry: the manually obtained original geometry (ground truth), the geometry reconstructed using the proposed CNN-based approach and the mean geometry specified in Eq. 2 (serving as the baseline). For each of the four subjects we have measured the end of diastole volume V^* and 24 circumferential strains, ε_i^*, $i = 1,\mathrm{K}\,,24$, from CMR scans, using procedures discussed in Section 2.1.3.

We carry out two studies: a real data study and a synthetic data study, each evaluated using the corresponding methods discussed in Section 2.5.2 (see also the bottom panel in Figure 7). In the *real data* study we use the measurements V^* and ε_i^*, $i = 1,\mathrm{K}\,,24$, available for each HV. However, we donot know the underlying true parameters with which we could compare the estimated values, therefore we treat the estimates from the original LV geometries as gold standards. These gold standard parameter values are also used to generate *synthetic data*: for each LV geometry we run the forward simulator with the corresponding optimized parameter values and treat the outputted measurements as synthetic data on the LV volume and 24 circumferential strains. The knowledge of the GT parameters allows us to assess the effect that the geometry approximations (the proposed CNN-based reconstruction and the mean geometry) have on the

optimization directly in parameter space.

3.4.1. Real data

Objective function. Table 5 presents the final values of the *loss* objective function from Eq. (7) for the four LV geometries under consideration. In all cases except HV B the reconstructed meshes lead to lower errors than the mean mesh. Moreover, the results seem stable, with independent runs of BO, even with the two different surrogates described in Section 2.5.1, leading to almost identical values.

Table 5: Bayesian optimization, real data study: the lowest values of the objective function for four healthy volunteers (HVs) obtained with the reconstructed geometry (geo.), the original geometry and the mean geometry. BO repeated with two algorithms (target surrogates, targ. and partial error surrogates, part.) for basic uncertainty quantification.

Subject	Reconstructed geo.		Original geo.		Mean geo.	
	BO targ.	BO part.	BO targ.	BO part.	BO targ.	BO part.
HV A	0.0448	0.0453	0.0396	0.0396	0.0938	0.0950
HV B	0.0580	0.0600	0.0443	0.0442	0.0385	0.0382
HV C	0.0797	0.0799	0.0655	0.0658	0.1118	0.1120
HV D	0.0470	0.0470	0.0556	0.0548	0.0476	0.0476

Stretch-stress curves and parameter estimates. Figures 11 and 12 present the stretch-stress curves corresponding to the final $\boldsymbol{\theta}$ values from the three geometries (reconstructed, original, mean) for each HV under consideration. To reiterate, the stretch-stress curve is an alternative way of depicting the nonlinear myocardial stiffness along a specific direction within a range of stretch levels. A larger stress under the same stretch level suggests a higher stiffness. Thus by comparing the relative positions of the stretch-stress curves, we can assess nonlinear myocardial stiffness within a range of stretch levels. For example, the closer the two curves, the closer the material stiffness, and vice versa. For all the HVs except HV C the curves obtained for the CNN reconstructed geometries are closer to those for the original geometries than the curves obtained using the mean geometry. For HV C (Figure 12), the parameters obtained with BO with partial error surrogates for the reconstructed and for the mean geometry lead to much stiffer effects for higher stretches than those found with other methods. Specifically, much higher stress values can

be found when $\lambda_f > 1.15$ or $\lambda_s > 1.25$ when using the BO with partial error surrogate method for HV C, in other words, much higher myocardial stiffness inferred by that method.

Figure 11: Bayesian optimization, real data study: stretch-stress curves for HV A and HV B (in rows). Blue: BO with target surrogates (targ.), red: BO with partial error surrogates (part.), solid lines: reconstructed geometries (recon. mesh), dashed lines: original geometries (org. mesh), dotted lines: mean geometries (mean mesh). $\lambda_f = l / L$ is the uniaxial stretch along the myocyte direction, and $\lambda_s = l / L$ is the uniaxial stretch along the sheet direction, in which l and L are the current and reference lengths of the myocardial strip. σ_f and σ_s are corresponding stress values.

Figure 12: Bayesian optimization, real data study: stretch-stress curves for HV C and HV D. Blue: BO with target surrogates (targ.), red: BO with partial error surrogates (part.), solid lines: reconstructed geometries (recon. mesh), dashed lines: original geometries (org. mesh), dotted lines: mean geometries (mean mesh). $\lambda_f = l / L$ is the uniaxial stretch along the myocyte direction, and $\lambda_s = l / L$ is the uniaxial stretch along the sheet direction, in which l and L are the current and reference lengths of the myocardial strip. σ_f and σ_s are corresponding stress values.

We report the final parameter estimates used to generate the stretch-stress curves in Table 6. These results are visualized by means of Bland-Altman plots comparing the differences between the original and reconstructed geometries with the differences between the original and mean geometries in Appendix A. Overall, taking the estimates obtained for the original geometries as benchmarks, we can conclude that the CNN reconstructed geometries lead to better estimates than those recorded for the mean mesh.

Finally, we observe that θ_4 is often hard to identify, even when using the original geometry. For instance for HV A and HV D the estimates of θ_4 for the original geometry obtained with the two variants of BO vary noticeably. A similar behaviour can be noted for HV C, for both the CNN-reconstructed and mean geometry, or HV B for the mean geometry. Therefore, in our next study based on synthetic data we decide to fix the value of $\theta_4 = 1$ (i.e. no scaling of the

reference values).

Table 6: Bayesian optimization, real data study: final parameter estimates (parameters leading to the minimum value of the objective function) for four healthy volunteers (HVs) obtained with the reconstructed geometry (geo.), the original geometry and the mean geometry. BO repeated with two algorithms (target surrogates, targ. and partial error surrogates, part.) for basic uncertainty quantification.

Subject	Reconstructed geo.		Original geo.		Mean geo.	
	BO targ.	BO part.	BO targ.	BO part.	BO targ.	BO part.
HV A						
θ_1	0.932	0.800	1.429	1.434	0.163	0.304
θ_2	2.663	2.449	2.585	2.569	1.980	1.952
θ_3	0.113	0.104	0.150	0.132	0.120	0.136
θ_4	1.112	3.163	0.141	0.240	4.858	3.579
HV B						
θ_1	0.100	0.200	0.100	0.100	0.958	0.750
θ_2	1.909	1.845	0.950	1.301	0.907	1.234
θ_3	0.113	0.218	2.879	1.751	4.993	4.971
θ_4	5.000	4.360	4.657	4.702	0.598	0.121
HV C						
θ_1	1.125	1.173	1.150	1.358	0.957	1.138
θ_2	0.896	0.839	1.667	1.342	1.957	0.669
θ_3	5.000	4.631	0.108	0.790	0.100	4.985
θ_4	0.719	1.541	4.986	5.000	4.999	1.079
HV D						
θ_1	2.338	2.189	0.901	0.969	4.998	4.999
θ_2	0.948	1.151	1.447	1.492	0.271	0.260

θ_3	2.044	1.449	0.107	0.322	0.100	0.519
θ_4	0.142	0.112	2.180	0.545	0.129	0.114

3.4.2. Synthetic data

Objective function. Table 7 presents the final values of the *loss* objective function from Eq. (7) for the four HVs under consideration. Notice that, due to the absence of measurement errors in the synthetic data, the mismatch values are much lower than those found in Table 5. These values are particularly close to zero for the original geometries due to the absence of any geometry approximation.

Similar to the real data study, the reconstructed geometries lead to lower losses than the mean geometry for all cases except HV B. In addition, the choice of surrogate has little effect on the parameter estimates, which is also found in the real data case.

Table 7: Bayesian optimization, synthetic data study: the lowest values of the objective function ($\times 100$) for four healthy volunteers (HVs) obtained with the reconstructed geometry (geo.), the original geometry and the mean geometry. BO repeated with two algorithms (target surrogates, targ. and partial error surrogates, part.) for basic uncertainty quantification.

	Reconstructed geo.		Original geo,		Mean geo.	
Subject	BO targ.	BO part.	BO targ.	BO part.	BO targ.	BO part.
HV A	0.449	0.449	0.000	0.000	5.195	5.197
HV B	0.446	0.430	0.015	0.001	0.247	0.247
HV C	0.809	0.800	0.003	0.001	1.685	1.666
HV D	1.253	1.253	0.000	0.001	5.643	5.642

Parameter estimates. Table 8 and Figure 13 present the results in the parameter space using rRMSEs, as defined in (9). As expected, optimization based on the original geometry performs the best with very low rRMSEs. Moreover, parameter estimates obtained with the CNN reconstructed geometries are typically much more accurate than those obtained with the mean geometry, leading to lower rRMSE. For all HVs but HV C, the CNN reconstructed geometries clearly outperform the mean geometry for both BO variants and for HV B the reconstructed geometry, and lead to results

very close to those based on the original geometry. For HV C, comparing the objective function values from Table 7 with the rRMSEs in Table 8 reveals that the optimization problem is difficult due to local optima and ridges in the objective function. We can see that even though independent BO runs converge to very close values of the objective function (e.g. 0.809×10^{-2} and 0.800×10^{-2} for the reconstructed mesh) the corresponding parameters may result in very different rRMSEs (e.g. 0.6377 and 0.1362 for the reconstructed mesh). This illustrates the point made in Section 2.5.2 that evaluation in the parameter space alone might not be sufficiently informative and needs to be complemented with the analysis of stretch-stress curves.

Table 8: Bayesian optimization, synthetic data study: relative RMSEs from Eq. (9) between the parameters leading to the minimum value of the objective function and the GT parameter values for four healthy volunteers (HVs) obtained with the reconstructed geometry (geo.), the original geometry and the mean geometry. BO repeated with two algorithm (target surrogates, targ. and partial error surrogates, part.) for basic uncertainty quantification. Bold font: subjects for which the reconstructed mesh outperforms the mean mesh.

Subject	Reconstructed geo.		Original geo.		Mean geo.	
	BO targ.	BO part.	BO targ.	BO part.	BO targ.	BO part.
HV A	0.1999	0.1964	0.0138	0.0116	0.4763	0.4862
HV B	0.1325	0.0421	0.1706	0.0087	0.9992	1.0264
HV C	0.6377	0.1362	0.1352	0.0312	0.6567	0.1057
HV D	0.1490	0.1770	0.0270	0.0297	0.5789	0.5626

Figure 13: Bayesian optimization, synthetic data study: relative RMSEs from Eq. (9) between the parameters leading to the minimum value of the objective function and the GT parameter values for four healthy volunteers (HVs) obtained with the reconstructed geometry (Recon. mesh.), the original geometry (Org. mesh) and the mean geometry (Mean mesh). BO repeated with two algorithms (target surrogates, targ. and partial error surrogates, part.) for basic uncertainty quantification.

Stretch-stress curves. Figures 14 and 15 present the stretch-stress curves for the four HVs under

consideration to assess the agreement between the inferred myocardial stiffness and the GT stiffness along the myocyte and sheet directions by virtually stretching a myocardial strip again. In all cases the curves based on the original geometries almost perfectly match the GT ones, suggesting the inferred myocardial stiffness agrees well with the GT stiffness at different stretch levels. Specifically, for HV A, the curves based on the reconstructed mesh and the curves based on the mean mesh are in close agreement with the GT curve for small stretches (up to approx. $\lambda_f = 1.1$ and $\lambda_s = 1.15$). For higher stretches, however, the curves based on the reconstructed mesh are noticeably closer to the GT curve than those based on the mean mesh. For HV B we record an almost perfect agreement between the GT curves and those based on the reconstructed mesh, both for the stretches along the myocyte direction and along the sheet direction. As in the HV A case, the curves based on the mean mesh deviate considerably from the GT curve for high stretches, indicating a much softer myocardium inferred due to the very low stress for high stretches compared to the GT curves. Finally, there is a very close agreement between the curves obtained for the CNN reconstructed geometries and those for the original geometries and the GT curves. On the other hand, the curves predicted for the mean geometry deviate noticeably from the remaining curves, especially for higher stretches, which means very different material properties inferred by using the mean geometry.

Figure 14: Bayesian optimization, synthetic data study: stretch-stress curves for HV A and HV B (in rows). Left: stretches along the myocyte direction, right: stretches along the sheet direction. Blue: BO with target surrogates (targ.), red: BO with partial error surrogates (part.), solid lines: reconstructed geometries (recon. mesh), dashed lines: original geometries (org. mesh), dotted lines: mean geometries (mean mesh), solid black lines: ground truth values. Solid black line: ground-truth curves.

Figure 15: Bayesian optimization, synthetic data study: stretch-stress curves for HV C and HV D (in rows). Left: stretches along the myocyte direction, right: stretches along the sheet direction. Blue: BO with target surrogates (targ.), red: BO with partial error surrogates (part.), solid lines: reconstructed geometries (recon. mesh), dashed lines: original geometries (org. mesh), dotted lines: mean geometries (mean mesh), solid black lines: ground truth values. Solid black line: ground-truth curves.

4. Discussion

4.1. Summary

We have developed a two-stage Deep Learning framework for automatic prediction of the LV geometry directly from CMR cine images. The first stage of the framework, the segmentation network, provides accurate annotations of LV wall for both SA and LA images. The second stage of the framework, the geometry prediction network, uses the predicted segemenations from the first stage to reconstruct a 17k-dimensional LV geometry. We have shown that this two-stage approach leads to considerably more accurate LV geometry reconstructions than the previous approach from [19]. In addition, we have introduced further enhancements, such as aligning the LA images and GT LV geometry reconstructions, which allow for a substantial reduction in reconstruction error.

In the context of cardiac-mechanic parameter inference, we have demonstrated that the LV geometry reconstructions from our two-stage framework can attain results that are much closer to those obtained with the GT, manually segmented geometries, than the results attained by using the mean mesh, in the sense of leading to more comparable stress-stain curves or lower values of the *loss* objective function in Eq. (7)

4.2. Principal component analysis

There has recently been substantial interest in LV geometry representation for cardio-mechanical modelling by means of statistical dimension reduction techniques. In particular, methods based on principal component analysis (PCA) [42] have been especially popular. The study in [43] implemented PCA for the purpose of load free geometry estimation in the context of inferring the passive stiffness of the myocardium. The work in [44] considered a PCA-reduced geometry representing the LV in a deep neural network model. The neural network approximated the diastolic filling process with the aim to infer the mechanical behaviour of myocardium. Our work relates to this strand of literature as we apply PCA within our geometry reconstruction network. In our case, PCA provides an essential dimension reduction and regularization step to enable the application of convolutional neural networks to learning the LV geometry from CMR images when the training data are comparatively small (in the order of several hundred exemplars). The optimal number of principal components has been decided by cross-validation. Using more components allows extra flexibility but at increased risk of overfitting, while using fewer components fails to sufficiently explain the variation in the data. However, the optimal number of

principal components depends on the data set size and can be expected to increase as larger data sets may become available in the future. See also the related discussion in Section 4.4.

4.3. Applications and potential impact in the clinic

Statistical emulators. As argued in Section 2.5, solving the cardiac mechanic equations must be done numerically. However a single run of the associated forward simulator (see Figure 6) typically takes 8-15 minutes. During inference, forward simulations have to be carried out repeatedly as part of an iterative optimization procedure, leading to computational execution times of several days, which is not viable for a practical clinical decision support system. Consequently, there is currently much interest in surrogate models and statistical emulators, in which a substantial part of computations is performed in advance, prior to recording any subject specific measurements.

Recent proof of concept studies in the context of cardiac mechanical modelling have demonstrated that the computational complexity can be reduced by several orders of magnitude at negligible loss in accuracy [33, 34, 45]. However, those studies assumed that the LV shape, extracted from a CMR scan of a healthy volunteer, was fixed. For real clinical applications based on personalized medicine, variations of the LV shape have to be included in the emulator. Unfortunately, the exploration of the full geometry representation space, discussed in Section 2.1.2, would require the impossible task of building an emulator in 17k dimensional space. A solution to this problem is to consider a low dimensional representation of the LV geometry, in a space that can be more carefully explored during emulator training. For clinical purposes we also require an efficient method for obtaining a projection of a new geometry into this space, ideally requiring minimal manual intervention by a practitioner. Our two-stage method does exactly this, by allowing prediction of a low dimensional representation directly from CMR images.

Other applications. Accelerating and automating the process of LV geometry reconstruction is a critical step towards personalized medicine that will obviate inefficient manual CMR image processing for LV geometry extraction. Once reconstructed, the predicted LV geometries can serve multiple purposes, of which the most important from our perspective is numerically solving cardiac mechanics equations with the finite element method. Moreover, the predicted LV geometries are valuable for the analysis of a number of clinically relevant quantities, such as LV volume, wall thickness and cavity volume. Finally, they will provide clinicians with a promising 3D visualization tool.

In addition, our proposed approach delivers automatically obtained LV wall segmentation, both in SA and LA images. As we have demonstrated, the segmentation accuracy on the test set is high, over 80% for the wall and over 90% for the cavity. In case higher accuracy is required, the predicted segmentation can be used as initializations which can be further manually corrected.

4.4. Limitations

Performance against GT meshes. While the improvement over the mean geometry performance as a reference benchmark is encouraging, there is still a significant performance gap between the reconstructed and the GT geometries, both in terms of direct geometric features, and derived features related to cardio-mechanics (cardio-mechanic parameters and stretch-stress curves). This suggests that further work is required to achieve the ultimate objective of reliable and automated clinical decision support.

Limited data. The main challenge for future work is enlarging the training set size. The set of LV geometries available in our study is limited to about 200 exemplars. This does not bring out the full potential of CNNs (due to the need for restrictive regularization), which have been conceived for "big data" problems. Obtaining a dataset orders of magnitude larger than the current one is fundamentally difficult due to the excessive computational cost of reconstructing a single LV geometry.

Motion correction. Motion artefacts, resulting from both cardiac and respiratory cycles and involuntary patient movement, still remain a great challenge in CMR imaging [46, 47], in particular for acute-MI patients with shortness of breath. Accurate reconstruction of the LV geometry is critical for biomechanical studies [3, 1], thus motion correction is generally needed when constructing patient-specific geometries from CMR cine images. In this study, a rigid-body translation has been applied to realign the endocardial and epicardial boundaries from SA images to the three LA images, as in [4], so that the ventricular boundaries from the SA images overlap with wall boundaries in LA images. Other approaches, such as tracking the imaging plane throughout the cardiac cycle, and deformable image registration schemes would further reduce motion artefacts [46]. However, it would be challenging to apply those methods to our CMR images retrospectively, thus a simple rigid-body translation has been used. Currently, there may be confounding effects from incompletely corrected motion artefacts, which are likely to lead to spurious deformations of excessively skewed or bulged LV geometries [10]. Since machine learning has been successfully applied to CMR image analysis and interpretation [13], we expect

that incorporating machine-learning based motion correction algorithms into the developed segmentation network will help alleviate motion artefacts.

Stiffness estimation in diseased hearts. LV geometries for both healthy volunteers and MI patients can be predicted by the automatic CNN-based framework, while we have only applied the Bayesian inference scheme to four healthy volunteers for estimating myocardial stiffness. To estimate myocardial stiffness in MI patients, a generalised cardiac MI model would be needed to account for material heterogeneity in the remote functional region and infarction region [4], leading to increased model complexity. Usually, late-gadolinium enhanced images need to be integrated into the 3D LV model for modelling the MI region. This process is further complicated by the fact that direct measurements of ventricle pressure values are not possible due to the invasive nature of such procedures. Nevertheless, the Bayesian inference procedure presented in this study can be directly applied to other heart diseases which are associated with a global change in material property, for example cardiac hypertrophy [48, 20].

Bi-ventricular geometry prediction. In the present study, only the LV geometry has been directly learned from CMR images. By including both LV and RV (right-ventricular) geometries, there will be more geometrical features in the data that could potentially enhance the accuracy and reliability of CNN-based geometry prediction. For example, Duan et al. [18] combined a multi-task deep learning approach along with atlas propagation to segment short-axis CMR volumetric images by learning the segmentation and landmark localization simultaneously, and ventricular shape prior knowledge was applied to overcome image artefacts. In future work, we will explore the prediction of the bi-ventricular geometry directly from conventional CMR images by incorporating into our CNN prior knowledge on features such as ventricular shapes, geometrical landmarks and measured wall motions.

5. Conclusions

We have developed an automatic CNN-based framework for predicting LV geometries directly from CMR images, without the need to perform manual annotations. We have recorded a noticeably lower LV mesh reconstruction error than previous methods. We have also demonstrated that our predicted LV geometries perform closer to the original, ground-truth geometries than the LV mean mesh in the context of cardiac mechanic parameter inference. A low-dimensional representation of the LV geometry delivered as a by-product of our proposed framework has the potential to be a stepping stone towards patient-specific statistical emulation, as

necessary for applications in personalized medicine.

Availability of data and materials

The original CMR cine images are not available due to privacy considerations. However, pre-processed data (a subset of re-scaled and cropped CMR images as well as segmented images and LV geometries) and the code are available in the following Github repository: https://github.com/aborowska/LVgeometry-prediction.

Funding

We would like to thank the funding from EPSRC (EP/S030875/1, EP/S020950/1, EP/N014642/1, EP/R018634/1, EP/T017899/1). DH is supported by a grant from the Royal Society of Edinburgh, award no. 62335. CB acknowledges a grant from the British Heart Foundation (RE/18/6134217).

Authors' contributions

AB, HG and DH jointly designed the study and coordinated the project. LR developed and implemented the convolutional neural networks and ran the training simulations. AB adapted the Bayesian optimization procedures and carried out the cardio-mechanic inference. AL and LR evaluated the prediction accuracy of the proposed method and computed the DICE scores. DD dealt with the LV alignment problem and performed the Procrustes transformations. CB led the CMR imaging project and provided the CMR scans. XL provided key inputs into the cardio-physiological interpretation of the results and helped with the strategic prioritization of tasks. AB, HG, DH and LR wrote the manuscript. All authors proofread the manuscripts and approved its submission.

Appendix A. Additional results

Appendix A.1. Bayesian optimization: real data

Figures A.16 and A.17 visualize the results from Table 6 by means of Bland-Altman plots. They are based on taking the estimates obtained with the original meshes as benchmarks and calculating the differences between this benchmark and the estimates obtained with other mesh types, i.e. CNN-reconstructed meshes (blue markers) and the mean mesh (red markers). For each mesh type we take the "representative" estimates, i.e. average estimates between two BO runs (with different emulation schemes). Since the differences can be both positive and negative we can

see in Figure A.16 that on average there is hardly any difference between the performance of reconstructed meshes and the mean mesh (solid horizontal lines). The former, however, produce differences much more concentrated around 0 as can be seen by considerably narrower confidence bounds (dashed lines). This can be easily seen in Figure A.17, in which the absolute values of the differences against the benchmark are depicted. Here we can see that the mean *absolute* difference with respect to the benchmark is much narrower for the estimates obtained with reconstructed meshes compared with those obtained with the mean mesh.

Figure A.16: Bayesian optimization, real data study: Bland-Altman plot comparing the differences between the final "representative" estimates based on the original and the reconstructed meshes (blue), and the differences between the final estimates based on the original meshes and the mean mesh (red). Each symbol represents a different parameter (circle – θ_1, square – θ_2, triangle – θ_3, star – θ_4) so it occurs eight times, twice (blue, red) for each HV (A, B, C, D). The horizontal lines are the mean (solid) and ± 1.96 standard deviation (dashed). For each mesh type (original, reconstructed, mean) we take the average between two BO runs (with target and partial error emulation) as the "representative" estimate for the corresponding HV and mesh type.

Figure A.17: Bayesian optimization, real data study: Bland-Altman plot comparing the absolute values of the differences between the final "representative" estimates based on the original and the reconstructed meshes (blue), and the differences between the absolute values of the final estimates based on the original meshes and the mean mesh (red). Each symbol represents a different parameter (circle – θ_1, square – θ_2, triangle – θ_3, star – θ_4) so it occurs eight times, twice (blue, red) for each HV (A, B, C, D). The horizontal lines are the mean (solid) and q_L and q_U, where q_L and q_U are empirical 2.5 and 97.5 percentiles, respectively, of the absolute differences. For each mesh type (original, reconstructed, mean) we take the average between two BO runs (with target and partial error emulation) as the "representative" estimate for the corresponding HV and mesh type.

References

[1] R. Chabiniok, V. Y. Wang, M. Hadjicharalambous, L. Asner, J. Lee, M. Sermesant, E. Kuhl, A. A. Young, P. Moireau, M. P. Nash, D. Chapelle, D. A. Nordsletten, Multiphysics and multiscale modelling, data–model fusion and integration of organ physiology in the clinic: ventricular cardiac mechanics, Interface Focus 6 (2016) 20150083–25.

[2] P. Lamata, A. Cookson, N. Smith, Clinical Diagnostic Biomarkers from the Personalization of Computational Models of Cardiac Physiology, Annals of Biomedical Engineering 44 (2016) 46–57.

[3] K. Mangion, H. Gao, D. Husmeier, X. Luo, C. Berry, Advances in computational modelling for personalised medicine after myocardial infarction, Heart 104 (2018) 550–557.

[4] H. Gao, K. Mangion, D. Carrick, D. Husmeier, X. Luo, C. Berry, Estimating prognosis in patients with acute myocardial infarction using personalized computational heart models, Scientific Reports 7 (2017 a) 1.

[5] H. Gao, A. Aderhold, K. Mangion, X. Luo, D. Husmeier, C. Berry, Changes and classification in myocardial contractile function in the left ventricle following acute myocardial infarction, Journal of The Royal Society Interface 14 (2017 b) 20170203.

[6] H. Gao, H. Wang, C. Berry, X. Luo, B. E. Griffith, Quasi-static image-based immersed boundary-finite element model of left ventricle under diastolic loading, International journal for numerical methods in biomedical engineering 30 (2014) 1199–1222.

[7] Z. J. Wang, V. Y. Wang, C. P. Bradley, M. P. Nash, A. A. Young, J. J. Cao, Left ventricular diastolic myocardial stiffness and end-diastolic myofibre stress in human heart failure using personalised biomechanical analysis, Journal of Cardiovascular Translational Research 11 (2018) 346–356.

[8] H. M. Wang, H. Gao, X. Y. Luo, C. Berry, B. E. Griffith, R. W. Ogden, T. J. Wang, Structure based finite strain modelling of the human left ventricle in diastole, International Journal for Numerical Methods in Biomedical Engineering 29 (2013) 83–103.

[9] G. A. Bello, T. J. Dawes, J. Duan, C. Biffi, A. de Marvao, L. S. Howard, J. S. R. Gibbs, M. R. Wilkins, et al., Deep-learning cardiac motion analysis for human survival prediction, Nature machine intelligence 1 (2019) 95.

[10] P. Medrano-Gracia, B. R. Cowan, D. A. Bluemke, J. P. Finn, J. A. Lima, A. Suinesiaputra, A. A. Young, Large scale left ventricular shape atlas using automated model fitting to

contours, in: International Conference on Functional Imaging and Modeling of the Heart, Springer, 2013, pp. 433–441.

[11] V. Wang, P. Nielsen, M. Nash, Image-based predictive modeling of heart mechanics, Annual review of biomedical engineering 17 (2015) 351–383.

[12] P. Lamata, M. Sinclair, E. Kerfoot, A. Lee, A. Crozier, B. Blazevic, S. Land, A. J. Lewandowski, D. Barber, S. Niederer, et al., An automatic service for the personalization of ventricular cardiac meshes, Journal of The Royal Society Interface 11 (2014) 20131023.

[13] T. Leiner, D. Rueckert, A. Suinesiaputra, B. Baeßler, R. Nezafat, I. Išgum, A. A. Young, Machine learning in cardiovascular magnetic resonance: basic concepts and applications, Journal of Cardiovascular Magnetic Resonance 21 (2019) 1–14.

[14] W. Bai, M. Sinclair, G. Tarroni, O. Oktay, M. Rajchl, G. Vaillant, A. M. Lee, N. Aung, E. Lukaschuk, M. M. Sanghvi, et al., Automated cardiovascular magnetic resonance image analysis with fully convolutional networks, Journal of Cardiovascular Magnetic Resonance 20 (2018) 65.

[15] X. Chen, B. M. Williams, S. R. Vallabhaneni, G. Czanner, R. Williams, Y. Zheng, Learning active contour models for medical image segmentation, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 11624–11632. doi: 10.1109/CVPR.2019.01190.

[16] F. Guo, M. Ng, M. Goubran, S. E. Petersen, S. K. Piechnik, S. Neubauer, G. Wright, Improving cardiac mri convolutional neural network segmentation on small training datasets and dataset shift: A continuous kernel cut approach, Medical Image Analysis 61 (2020) 101636.

[17] O. Bernard, A. Lalande, C. Zotti, F. Cervenansky, X. Yang, P. A. Heng, I. Cetin, K. Lekadir, O. Camara, M. A. Gonzalez Ballester, G. Sanroma, S. Napel, S. Petersen, G. Tziritas, E. Grinias, M. Khened, V. A. Kollerathu, G. Krishnamurthi, M. M. Rohé, X. Pennec, M. Sermesant, F. Isensee, P. Jäger, K. H. Maier-Hein, P. M. Full, I. Wolf, S. Engelhardt, C. F. Baumgartner, L. M. Koch, J. M. Wolterink, I. Išgum, Y. Jang, Y. Hong, J. Patravali, S. Jain, O. Humbert, P. M. Jodoin, Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved?, IEEE Transactions on Medical Imaging 37 (2018) 2514–2525.

[18] J. Duan, G. Bello, J. Schlemper, W. Bai, T. J. Dawes, C. Biffi, A. de Marvao, G. Doumoud,

D. P. O'Regan, D. Rueckert, Automatic 3d bi-ventricular segmentation of cardiac images by a shape-refined multi-task deep learning approach, IEEE transactions on medical imaging 38 (2019) 2151–2164.

[19] L. Romaszko, A. Borowska, A. Lazarus, H. Gao, X. Luo, D. Husmeier, Direct learning left ventricular meshes from CMR images, International Conference on Statistics: Theory and Applications 2019 (2019).

[20] W. Li, A. Lazarus, H. Gao, A. Martinez-Naharro, M. Fontana, P. Hawkins, S. Biswas, R. Janiczek, J. Cox, C. Berry, et al., Analysis of cardiac amyloidosis progression using model-based markers, Frontiers in Physiology 11 (2020) 324.

[21] K. Mangion, H. Gao, C. McComb, D. Carrick, G. Clerfond, X. Zhong, X. Luo, C. E. Haig, C. Berry, A Novel Method for Estimating Myocardial Strain: Assessment of Deformation Tracking Against Reference Magnetic Resonance Methods in Healthy Volunteers, Scientific Reports 6 (2016) 38774.

[22] D. Carrick, K. Oldroyd, M. McEntegart, C. Haig, M. Petrie, H. Eteiba, S. Hood, C. Owens, S. Watkins, J. Layland, et al., A randomized trial of deferred stenting versus immediate stenting to prevent no-or slow-reflow in acute ST-segment elevation myocardial infarction (DEFER-STEMI), Journal of the American College of Cardiology 63 (2014) 2088–2098.

[23] Y. Liu, H. Wen, R. C. Gorman, V. Pilla, J. H. Gorman III, G. Buckberg, S. D. Teague, G. S. Kassab, Reconstruction of myocardial tissue motion and strain fields from displacement-encoded MR imaging, American Journal of Physiology-Heart and Circulatory Physiology 297 (2009) H1151–H1162.

[24] H. Gao, W. G. Li, L. Cai, C. Berry, X. Y. Luo, Parameter estimation in a Holzapfel–Ogden law for healthy myocardium, Journal of Engineering Mathematics 95 (2015) 231–248.

[25] M. D. Cerqueira, N. J. Weissman, V. Dilsizian, A. K. Jacobs, S. Kaul, W. K. Laskey, D. J. Pennell, J. A. Rumberger, T. Ryan, M. S. Verani, Standardized myocardial segmentation and nomenclature for tomographic imaging of the heart: a statement for healthcare professionals from the Cardiac Imaging Committee of the Council on Clinical Cardiology of the American Heart Association, Circulation 105 (2002) 539–542.

[26] F. Tonolini, J. Radford, A. Turpin, D. Faccio, R. Murray-Smith, Variational inference for computational imaging problems, arxiv:1904.0624v2 (2020).

[27] L. Romaszko, A. Lazarus, H. Gao, A. Borowska, X. Luo, D. Husmeier, Massive

dimensionality reduction for the left ventricular mesh, International Conference on Statistics: Theory and Applications 2019 (2019).

[28] C. M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1995.

[29] L. Perez, J. Wang, The effectiveness of data augmentation in image classification using deep learning, arXiv preprint arXiv:1712.04621 (2017).

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, The Journal of Machine Learning Research 15 (2014) 1929–1958.

[31] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.

[32] G. A. Holzapfel, R. W. Ogden, Constitutive modelling of passive myocardium: a structurally based framework for material characterization, Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 367 (2009) 3445–3475.

[33] V. Davies, U. Noè, A. Lazarus, H. Gao, B. Macdonald, C. Berry, X. Luo, D. Husmeier, Fast parameter inference in a biomechanical model of the left ventricle by using statistical emulation, Journal of the Royal Statistical Society: Series C (Applied Statistics) 68 (2019) 1555–1576.

[34] U. Noè, A. Lazarus, H. Gao, V. Davies, B. Macdonald, K. Manglion, C. Berry, X. Luo, D. Husmeier, Gaussian process emulation to accelerate parameter estimation in a mechanical model of the left ventricle: a critical step towards clinical end-user relevance, Journal of the Royal Society: Interface 16 (2019) 20190114.

[35] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, N. de Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proceedings IEEE 104 (2016) 148–175.

[36] A. Borowska, H. Gao, A. Lazarus, D. Husmeier, Bayesian optimisation for efficient parameter inference in a cardiac mechanics model of the left ventricle, submitted (2020).

[37] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning, MIT Press, Cambridge, MA, 2006.

[38] D. R. Jones, M. Schonlau, W. J. Welch, Efficient global optimization of expensive black-box functions, Journal of Global optimization 13 (1998) 455–492.

[39] G. Sommer, A. J. Schriefl, M. Andrä, M. Sacherer, C. Viertler, H. Wolinski, G. A. Holzapfel, Biomechanical properties and microstructure of human ventricular myocardium, Acta biomaterialia 24 (2015) 172–192.

[40] D. Guan, F. Ahmad, P. Theobald, S. Soe, X. Luo, H. Gao, On the AIC-based model reduction for the general Holzapfel–Ogden myocardial constitutive law, Biomechanics and Modeling in Mechanobiology 18 (2019) 1213–1232.

[41] F. Liao, X. Chen, X. Hu, S. Song, Estimation of the volume of the left ventricle from MRI images using deep neural networks, IEEE Transactions on Cybernetics 49 (2019) 495–504.

[42] J. Shlens, A tutorial on principal component analysis, 2014. arXiv:1404.1100.

[43] Z. J. Wang, V. Y. Wang, T. P. Babarenda Gamage, V. Rajagopal, J. J. Cao, P. M. F. Nielsen, C. P. Bradley, A. A. Young, M. P. Nash, Efficient estimation of load-free left ventricular geometry and passive myocardial properties using principal component analysis, International journal for numerical methods in biomedical engineering 36 (2020) e3313.

[44] G. D. Maso Talou, T. P. Babarenda Gamage, M. Sagar, M. P. Nash, Deep learning over reduced intrinsic domains for efficient mechanics of the left ventricle, Frontiers in Physics 8 (2020) 30.

[45] D. Dalton, D. Husmeier, Improved statistical emulation for a soft-tissue cardiac mechanical model, in: I. Irigoien, D.-J. Lee, J. Martínez-Minaya, M. X. Rodríguez-Álvarez (Eds.), Proceedings of the 35th International Workshop on Statistical Modelling, volume 1, Servicio Editorial de la Universidad del País Vasco, Bilbao, Spain, 2020, pp. 55–60.

[46] A. D. Scott, J. Keegan, D. N. Firmin, Motion in cardiovascular mr imaging, Radiology 250 (2009) 331–351.

[47] M. Salerno, B. Sharif, H. Arheden, A. Kumar, L. Axel, D. Li, S. Neubauer, Recent advances in cardiovascular magnetic resonance: techniques and applications, Circulation: Cardiovascular Imaging 10 (2017) e003951.

[48] K. Yamamoto, T. Masuyama, Y. Sakata, N. Nishikawa, T. Mano, J. Yoshida, T. Miwa, M. Sugawara, Y. Yamaguchi, T. Ookawara, et al., Myocardial stiffness is determined by ventricular fibrosis, but not by compensatory or excessive hypertrophy in hypertensive heart, Cardiovascular research 55 (2002) 76–82.

Declaration of competing interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Highlights

Automatic CNN-based framework for predicting LV geometries directly from CMR images
Two-stage approach combines a segmentation network and geometry prediction network
Geometry prediction based on principal component analysis for dimensionality reduction
CNN-predicted LV geometries useful in biomechanical studies of LV stiffness estimation
Stepping stone towards the development of LV geometry-heterogeneous emulators

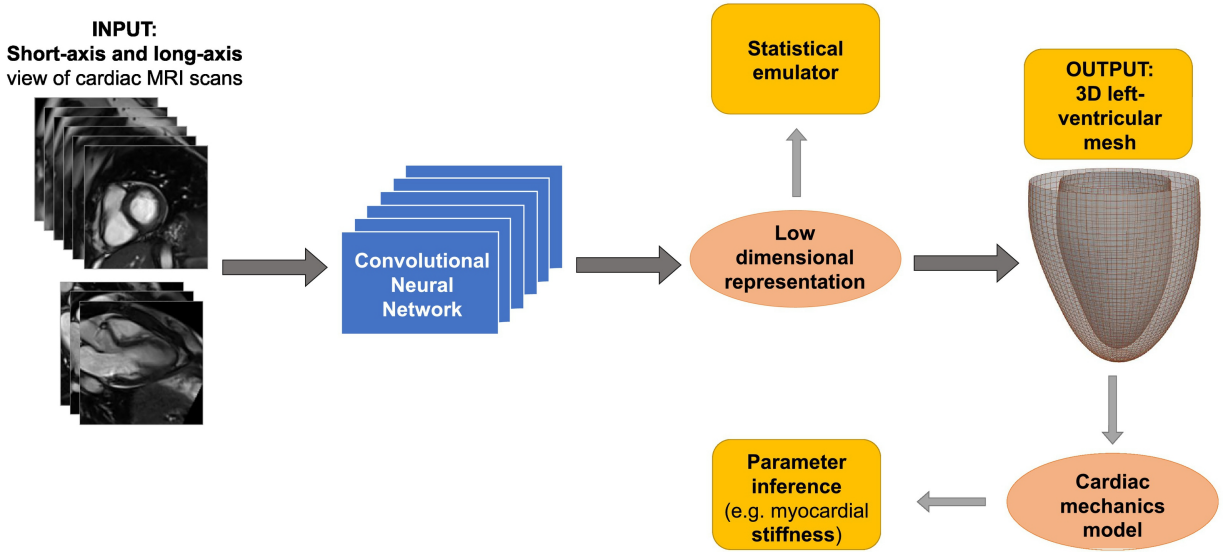
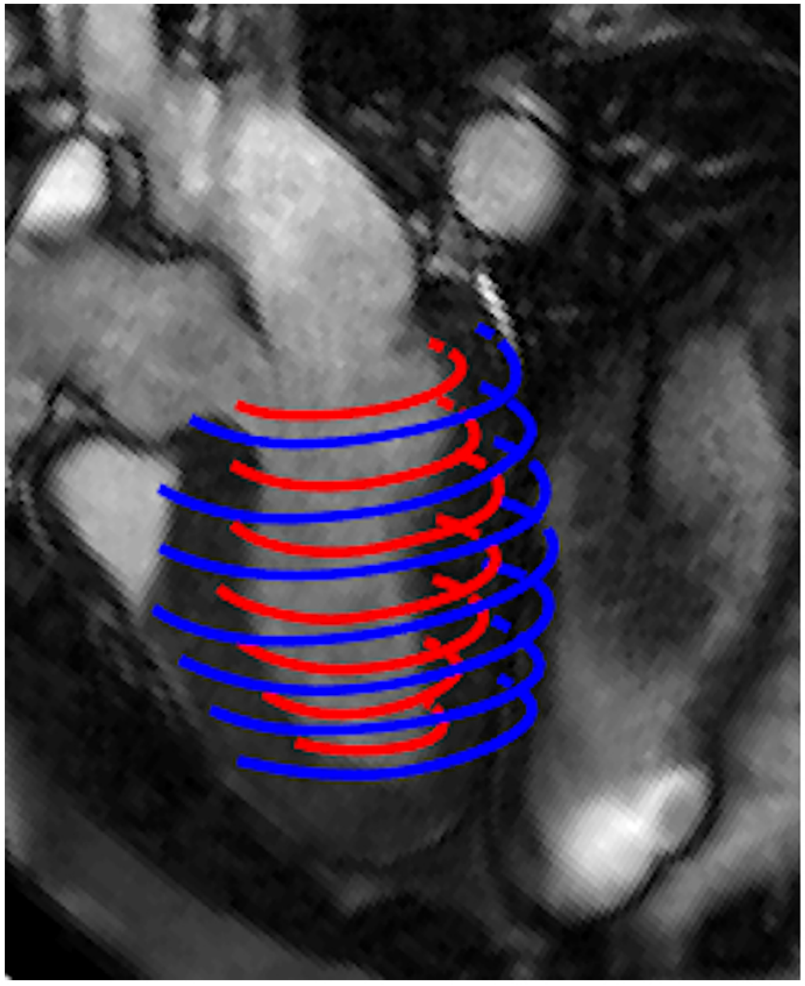

INPUT:
Short-axis and long-axis
view of cardiac MRI scans
Convolutional
Neural
Network
Low
dimensional
representation
Statistical
emulator
OUTPUT:
3D left-
ventricular
mesh
Cardiac
mechanics
model
Parameter
inference
(e.g. myocardial
stiffness)

Figure 1



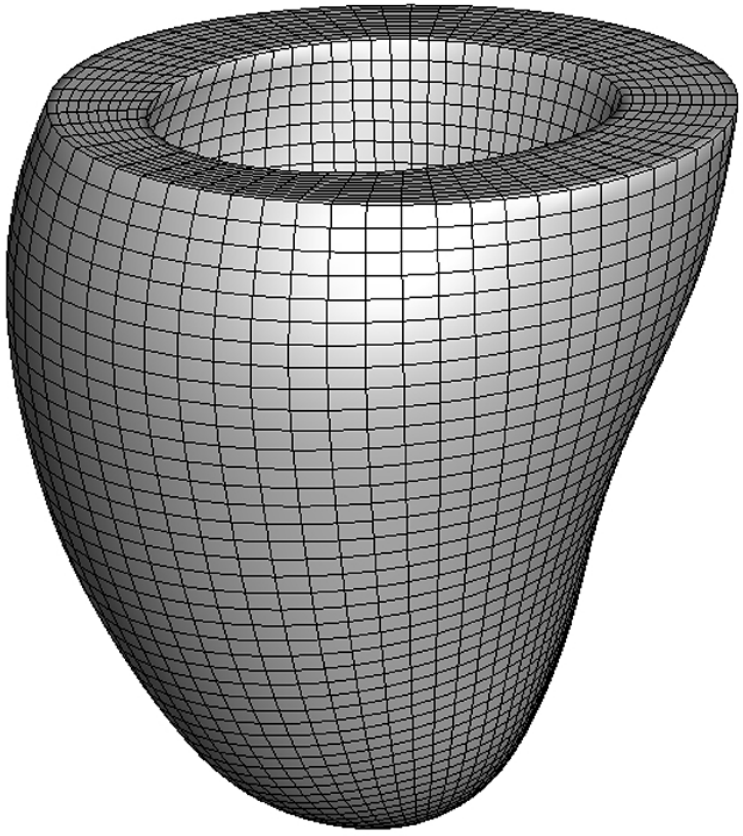


Figure 2

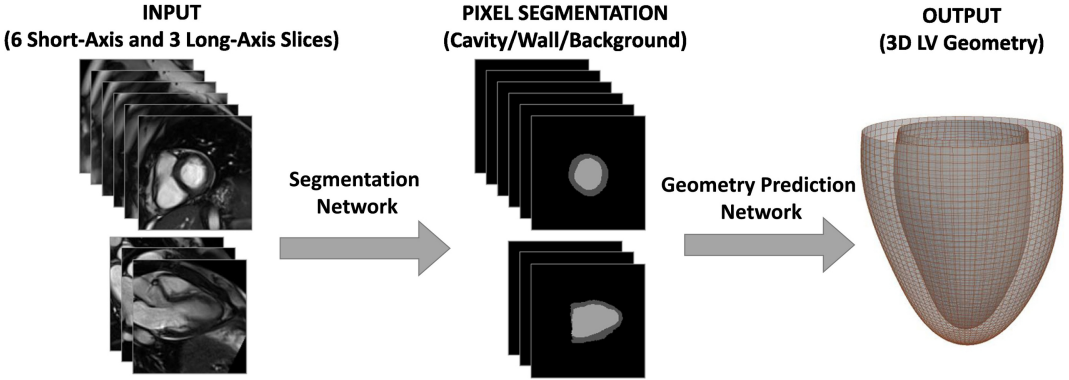

INPUT
(6 Short-Axis and 3 Long-Axis Slices)
Segmentation Network
PIXEL SEGMENTATION
(Cavity/Wall/Background)
Geometry Prediction Network
OUTPUT
(3D LV Geometry)

Figure 3

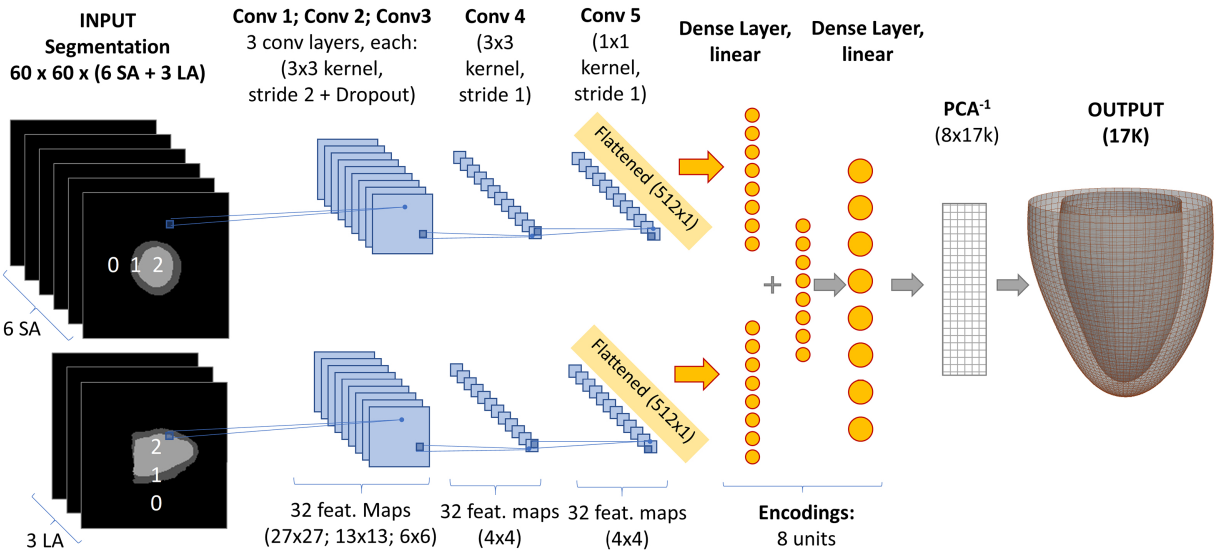

INPUT
Segmentation
60 x 60 x (6 SA + 3 LA)
Conv 1; Conv 2; Conv3
3 conv layers, each:
(3x3 kernel,
stride 2 + Dropout)
Conv 4
(3x3
kernel,
stride 1)
Conv 5
(1x1
kernel,
stride 1)
Dense Layer,
linear
Dense Layer,
linear
PCA-1
(8x17k)
OUTPUT
(17K)
0 1 2
6 SA
2
1
0
3 LA
Flattened (512x1)
Flattened (512x1)
+
32 feat. Maps
(27x27; 13x13; 6x6)
Dropout=0.01
32 feat. maps
(4x4)
32 feat. maps
(4x4)
Encodings:
8 units

Figure 4

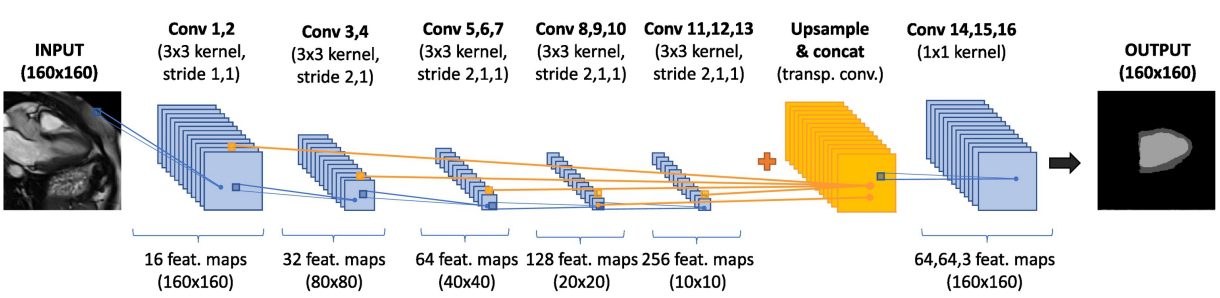

INPUT
(160x160)
Conv 1,2
(3x3 kernel,
stride 1,1)
Conv 3,4
(3x3 kernel,
stride 2,1)
Conv 5,6,7
(3x3 kernel,
stride 2,1,1)
Conv 8,9,10
(3x3 kernel,
stride 2,1,1)
Conv 11,12,13
(3x3 kernel,
stride 2,1,1)
Upsample
& concat
(transp. conv.)
Conv 14,15,16
(1x1 kernel)
OUTPUT
(160x160)
16 feat. maps
(160x160)
32 feat. maps
(80x80)
64 feat. maps
(40x40)
128 feat. maps
(20x20)
256 feat. maps
(10x10)
64,64,3 feat. maps
(160x160)

Figure 5

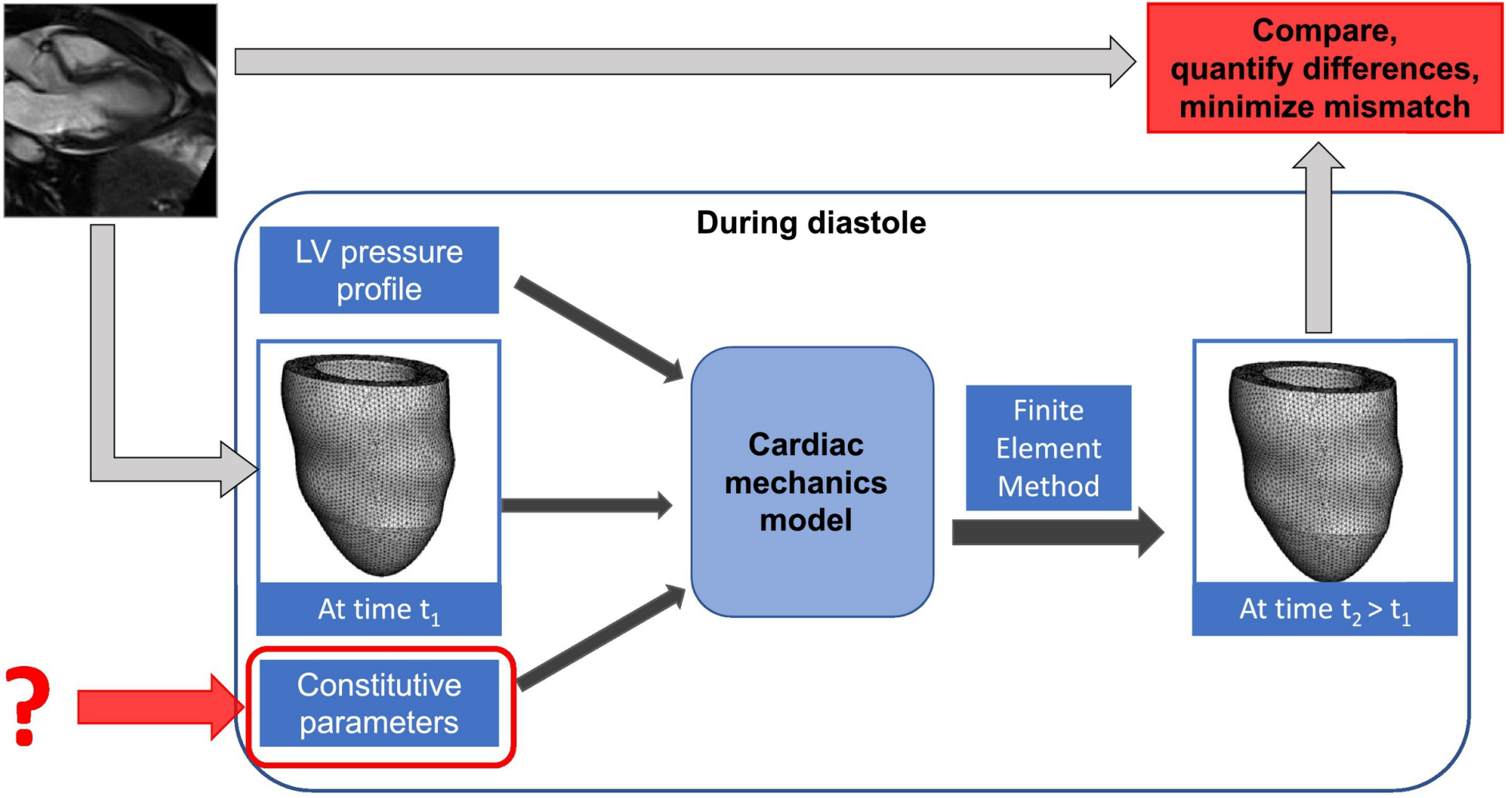

Compare, quantify differences, minimize mismatch
During diastole
LV pressure profile
At time t_1
Constitutive parameters
?
Cardiac mechanics model
Finite Element Method
At time $t_2 > t_1$

Figure 6

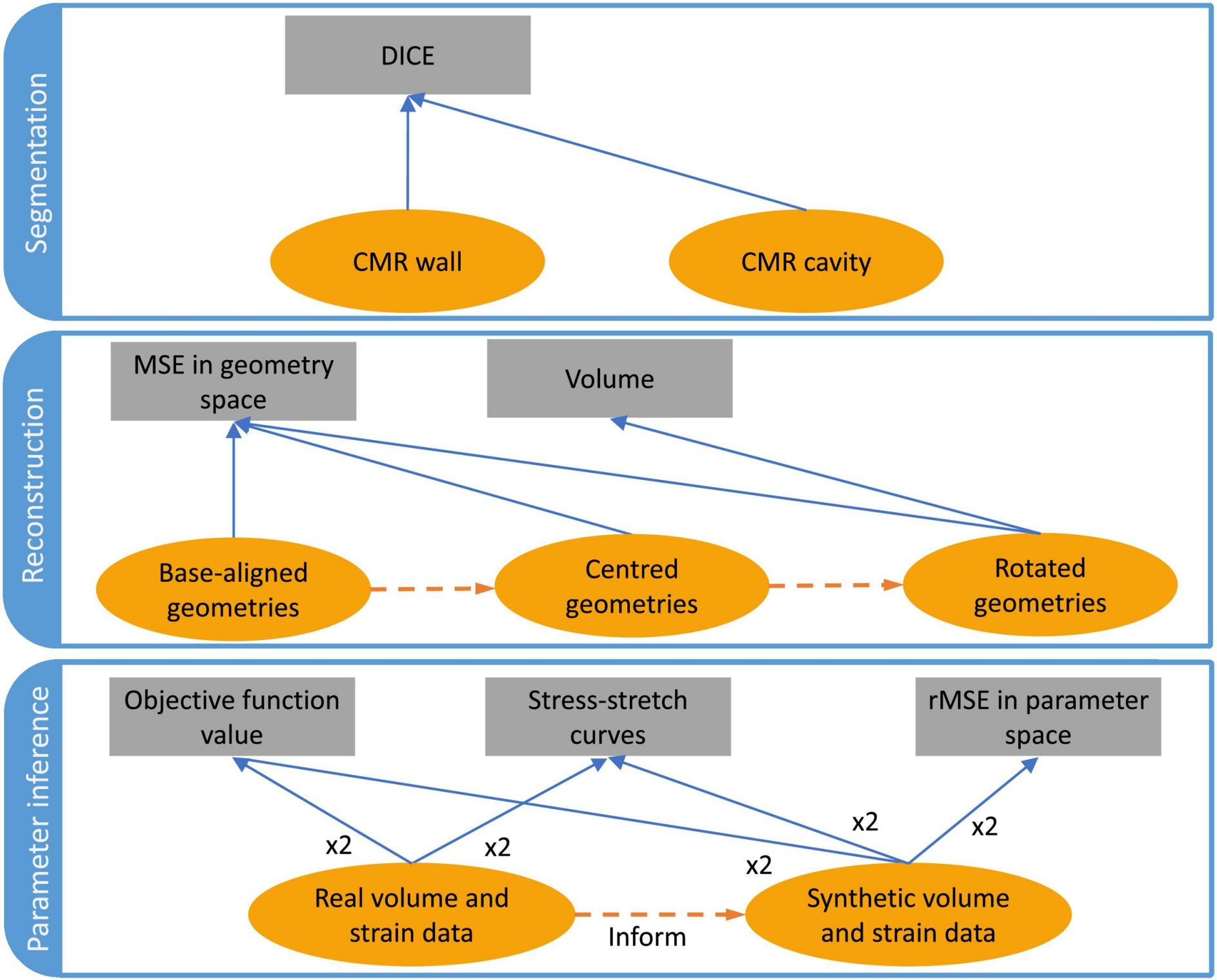

Segmentation
DICE
CMR wall
CMR cavity
Reconstruction
MSE in geometry space
Volume
Base-aligned geometries
Centred geometries
Rotated geometries
Parameter inference
Objective function value
Stress-stretch curves
rMSE in parameter space
x2
x2
x2
x2
Real volume and strain data
Inform
Synthetic volume and strain data

Figure 7

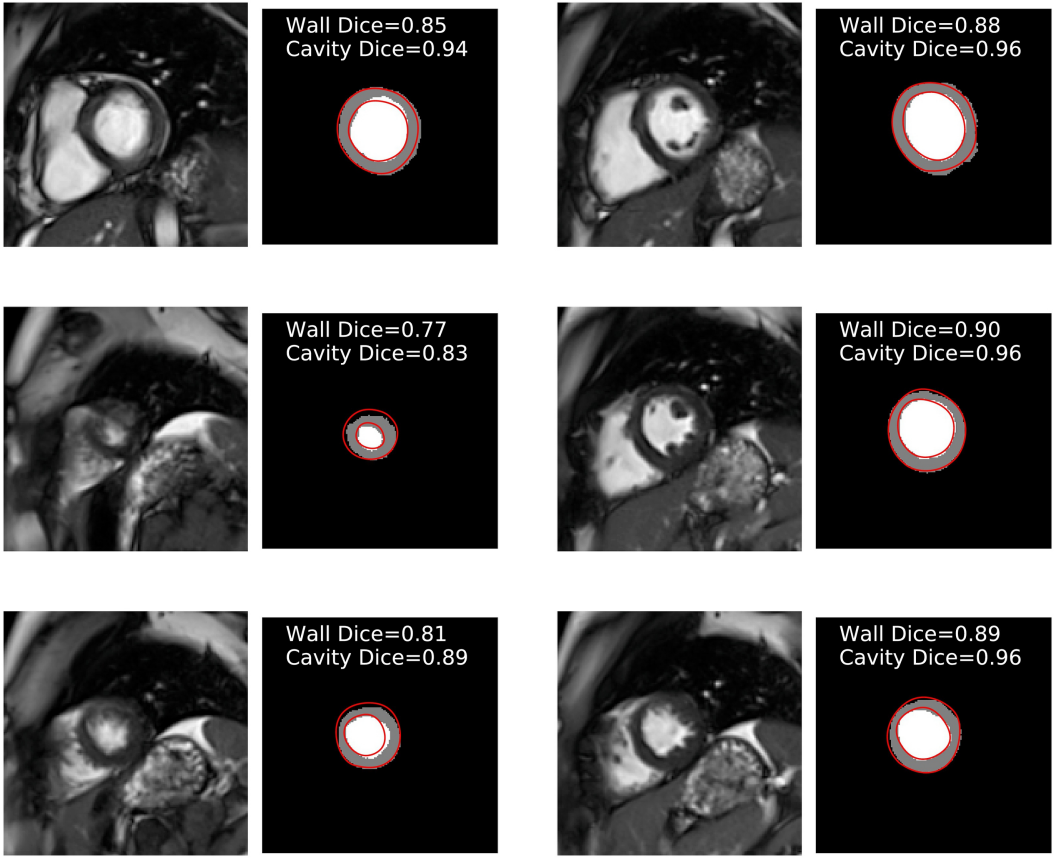

Wall Dice=0.85
Cavity Dice=0.94
Wall Dice=0.88
Cavity Dice=0.96
Wall Dice=0.77
Cavity Dice=0.83
Wall Dice=0.90
Cavity Dice=0.96
Wall Dice=0.81
Cavity Dice=0.89
Wall Dice=0.89
Cavity Dice=0.96

Figure 8

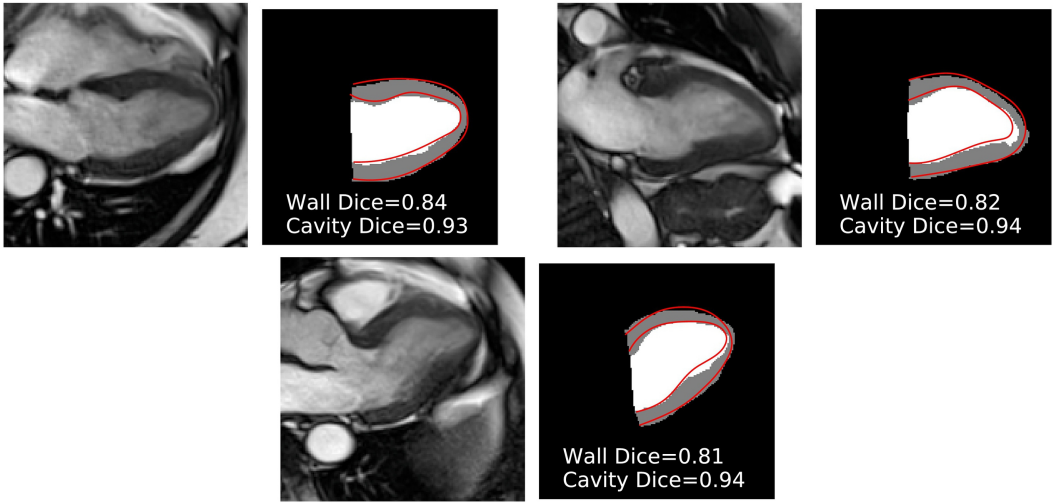

Wall Dice=0.84
Cavity Dice=0.93
Wall Dice=0.82
Cavity Dice=0.94
Wall Dice=0.81
Cavity Dice=0.94

Figure 9

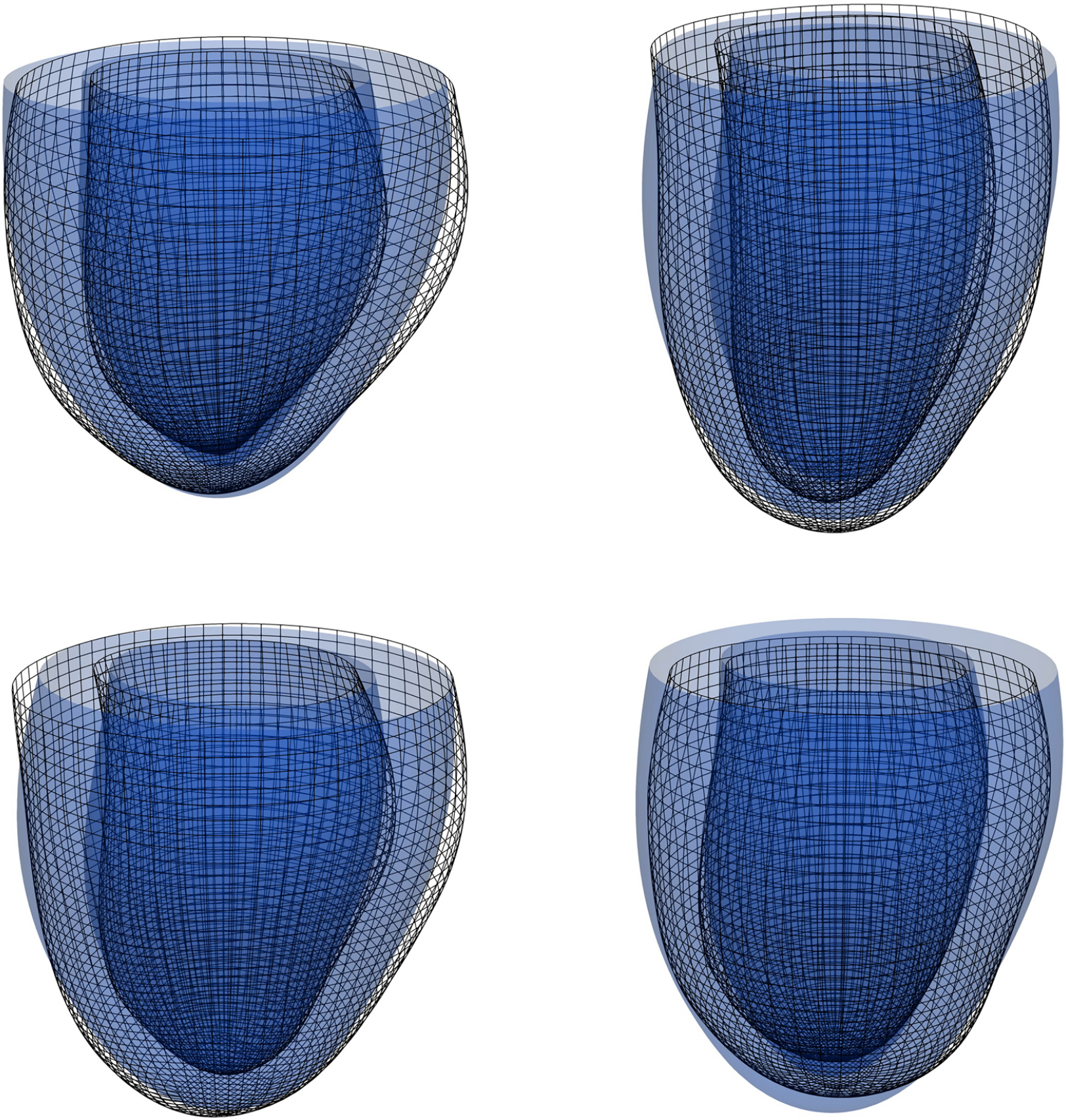


Figure 10

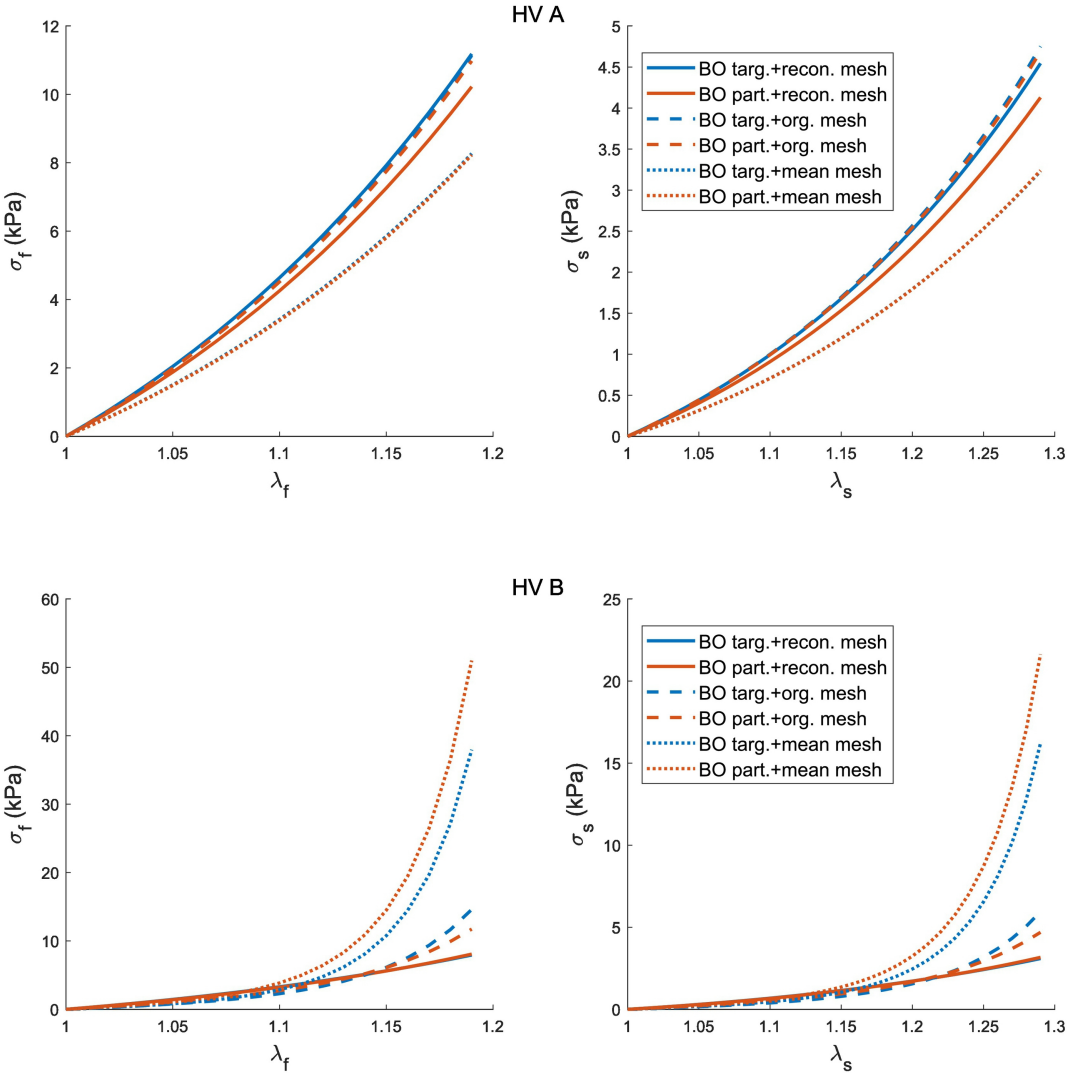

HV A
HV B
BO targ.+recon. mesh
BO part.+recon. mesh
BO targ.+org. mesh
BO part.+org. mesh
BO targ.+mean mesh
BO part.+mean mesh
σf (kPa)
σs (kPa)
λf
λs

Figure 11

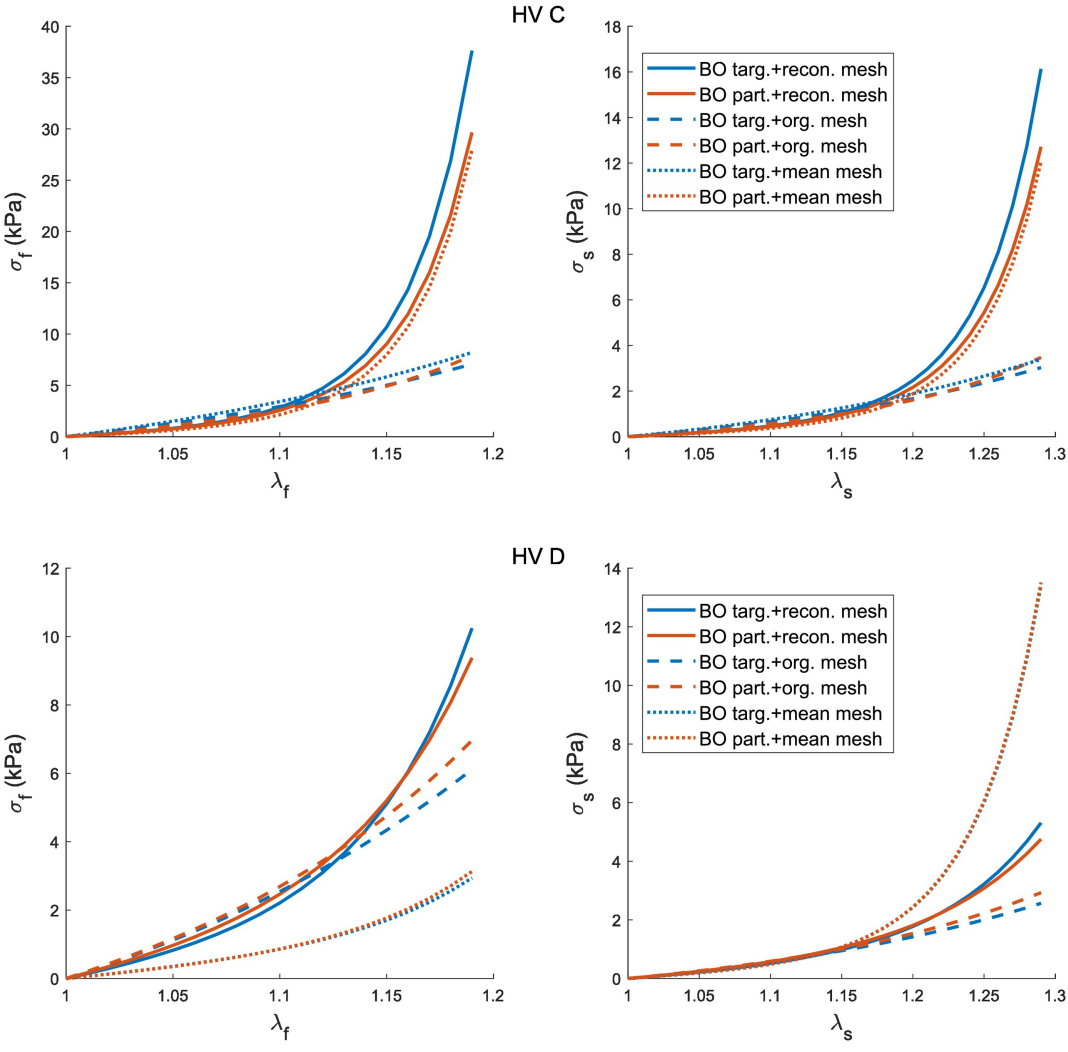

HV C
σ_f (kPa)
λ_f
σ_s (kPa)
λ_s
BO targ.+recon. mesh
BO part.+recon. mesh
BO targ.+org. mesh
BO part.+org. mesh
BO targ.+mean mesh
BO part.+mean mesh
HV D
σ_f (kPa)
λ_f
σ_s (kPa)
λ_s
BO targ.+recon. mesh
BO part.+recon. mesh
BO targ.+org. mesh
BO part.+org. mesh
BO targ.+mean mesh
BO part.+mean mesh

Figure 12

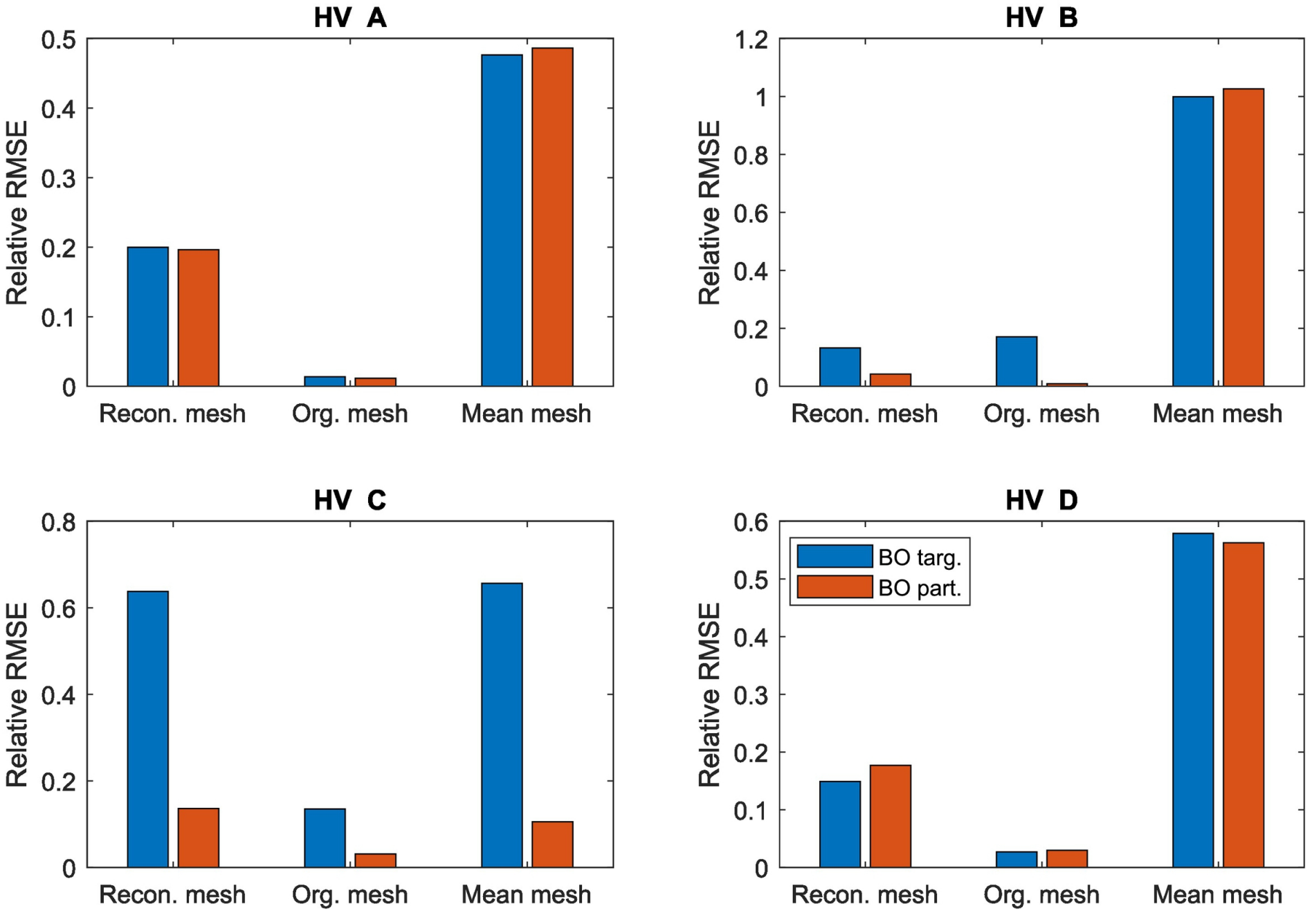

HV A
Relative RMSE
0.5
0.4
0.3
0.2
0.1
0
Recon. mesh
Org. mesh
Mean mesh
HV B
Relative RMSE
1.2
1
0.8
0.6
0.4
0.2
0
Recon. mesh
Org. mesh
Mean mesh
HV C
Relative RMSE
0.8
0.6
0.4
0.2
0
Recon. mesh
Org. mesh
Mean mesh
HV D
Relative RMSE
0.6
0.5
0.4
0.3
0.2
0.1
0
BO targ.
BO part.
Recon. mesh
Org. mesh
Mean mesh

Figure 13

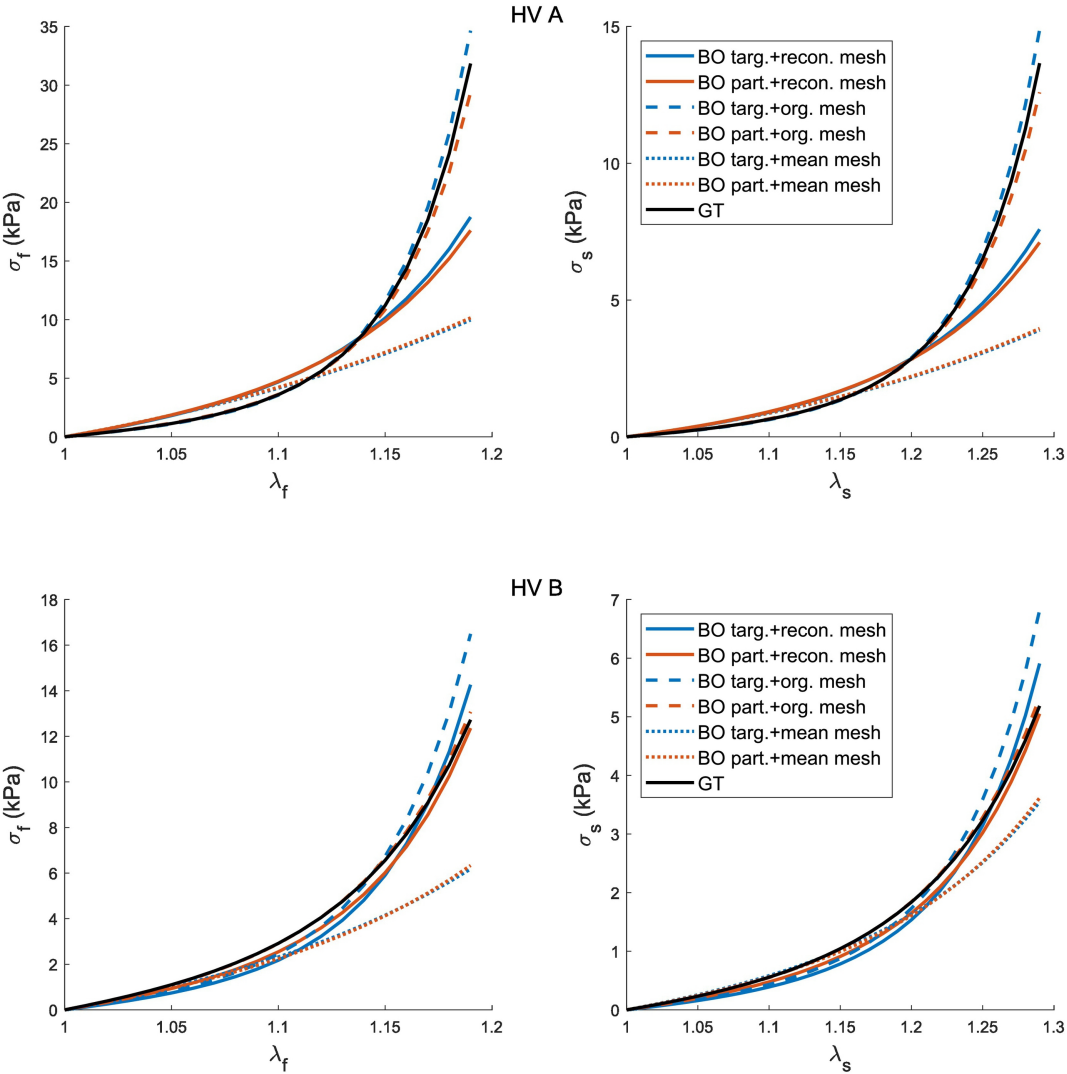

HV A
σ_f (kPa)
λ_f
σ_s (kPa)
λ_s
BO targ.+recon. mesh
BO part.+recon. mesh
BO targ.+org. mesh
BO part.+org. mesh
BO targ.+mean mesh
BO part.+mean mesh
GT
HV B
σ_f (kPa)
λ_f
σ_s (kPa)
λ_s
BO targ.+recon. mesh
BO part.+recon. mesh
BO targ.+org. mesh
BO part.+org. mesh
BO targ.+mean mesh
BO part.+mean mesh
GT

Figure 14

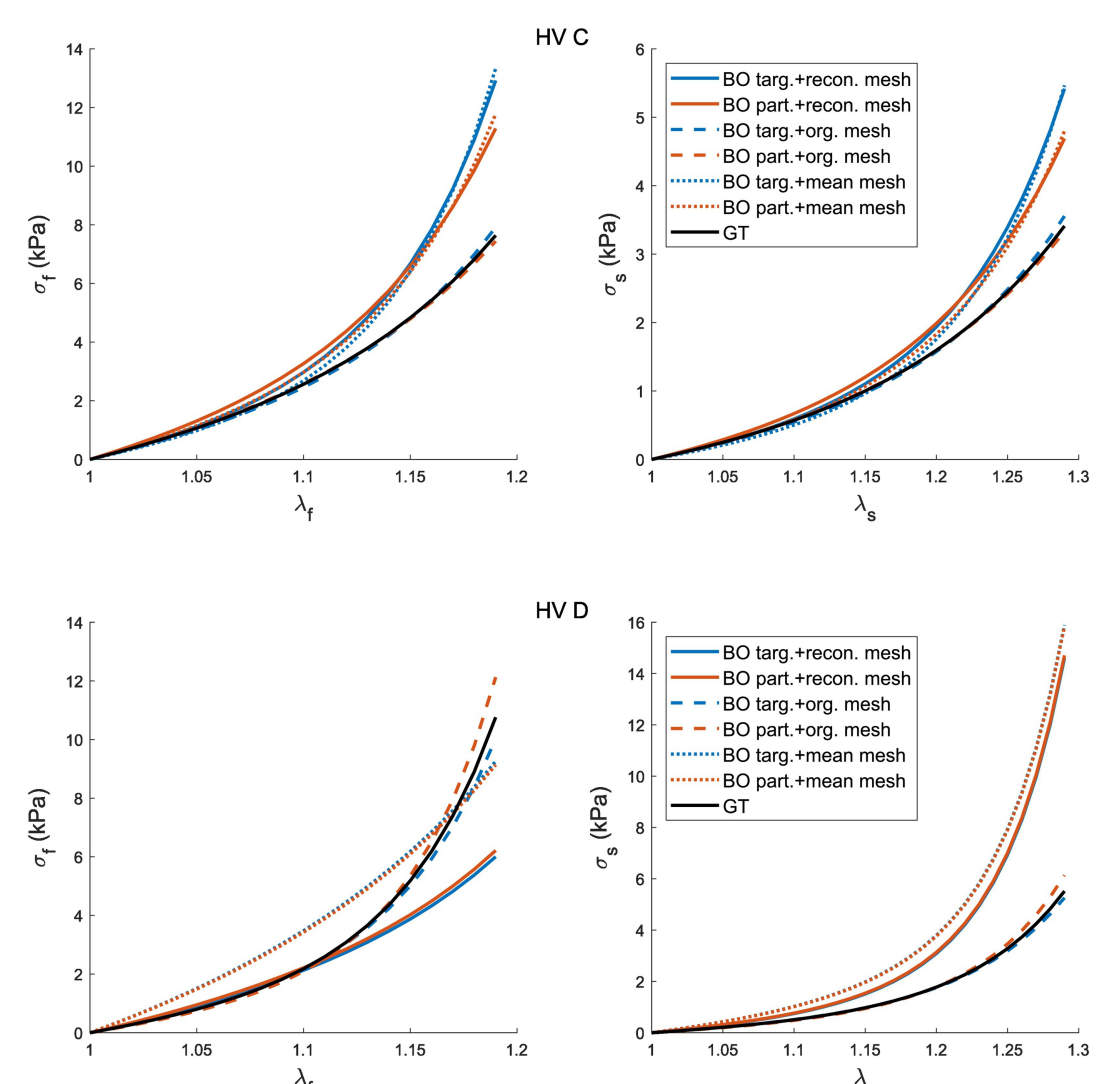

HV C
HV D
BO targ.+recon. mesh
BO part.+recon. mesh
BO targ.+org. mesh
BO part.+org. mesh
BO targ.+mean mesh
BO part.+mean mesh
GT
σ_f (kPa)
σ_s (kPa)
λ_f
λ_s

Figure 15