

# A machine learning approach as a surrogate of finite element analysis–based inverse method to estimate the zero-pressure geometry of human thoracic aorta

Liang Liang | Minliang Liu | Caitlin Martin | Wei Sun 

Tissue Mechanics Laboratory, The  
Wallace H. Coulter Department of  
Biomedical Engineering, Georgia Institute  
of Technology and Emory University,  
Atlanta, GA, USA

## Correspondence

Wei Sun, The Wallace H. Coulter  
Department of Biomedical Engineering,  
Georgia Institute of Technology and  
Emory University, Technology Enterprise  
Park, Room 206, 387 Technology Circle  
NW, Atlanta, GA 30313-2412, USA.  
Email: wei.sun@bme.gatech.edu

## Funding information

American Heart Association, Grant/  
Award Number: 16POST30210003; NIH,  
National Heart, Lung, and Blood Institute,  
Grant/Award Numbers: R21 HL127570  
and R01 HL104080

## Abstract

Advances in structural finite element analysis (FEA) and medical imaging have made it possible to investigate the in vivo biomechanics of human organs such as blood vessels, for which organ geometries at the zero-pressure level need to be recovered. Although FEA-based inverse methods are available for zero-pressure geometry estimation, these methods typically require iterative computation, which are time-consuming and may be not suitable for time-sensitive clinical applications. In this study, by using machine learning (ML) techniques, we developed an ML model to estimate the zero-pressure geometry of human thoracic aorta given 2 pressurized geometries of the same patient at 2 different blood pressure levels. For the ML model development, a FEA-based method was used to generate a dataset of aorta geometries of 3125 virtual patients. The ML model, which was trained and tested on the dataset, is capable of recovering zero-pressure geometries consistent with those generated by the FEA-based method. Thus, this study demonstrates the feasibility and great potential of using ML techniques as a fast surrogate of FEA-based inverse methods to recover zero-pressure geometries of human organs.

## KEYWORDS

finite element analysis, machine learning, neural network, zero-pressure geometry

## 1 | INTRODUCTION

Structural nonlinear finite element analysis (FEA) has been increasingly used to investigate the biomechanics of human organs, such as heart valves,<sup>1</sup> ventricles,<sup>2</sup> and blood vessels,<sup>3–5</sup> for which in vivo organ geometries can be obtained from 3D imaging systems (eg, computed tomography). Since nonlinear FEA usually starts from the stress-free state and the organs in in vivo images are often at a loaded state, organ geometries at the zero-pressure level are needed, serving as a good approximation of the stress-free state, for accurate stress analysis at loaded states.<sup>5,6</sup> Methods to obtain zero-pressure geometries may have potential applications well beyond FEA. Currently, by using 3D printing technology, a phantom model of a human organ can be built to have a patient-specific geometry reconstructed from 3D images<sup>7</sup> and similar mechanical properties of biological tissue.<sup>8,9</sup> Such 3D-printed models, if built with accurate zero-pressure geometries, would enable more realistic in vitro simulation platforms for preoperative planning, such as in vitro simulation of transcatheter aortic valve replacement.<sup>8</sup>

In the literature, several methods<sup>10–20</sup> are available to estimate zero-pressure organ geometries. These methods usually take as input a loaded geometry reconstructed from in vivo images and assume the material properties are known,

then either (1) estimate the zero-pressure geometry<sup>10,14,16-18,20</sup> by adjusting a candidate geometry and running forward finite element (FE) simulations, or (2) estimate the prestress/prestrain field on the loaded configuration,<sup>12,13,15,19,21</sup> which can be used to back out the zero-pressure geometry (eg, by depressurizing the FE model), or (3) estimate the zero-pressure geometry and the prestress field using an inverse FE formulation.<sup>11</sup> All of these methods rely on FEA and most of them require many iterations of FE simulations, which makes these methods very time-consuming and inefficient. Furthermore, since FEA may often encounter numerical convergence issues, especially with nonlinear tissue mechanical properties, which often require experienced human experts to resolve through trial and error, ie, to adjust the parameters of FE solvers or the FE models, these methods can be highly time-consuming and costly. Thus, there is a need for robust and efficient algorithms to obtain zero-pressure organ geometries.

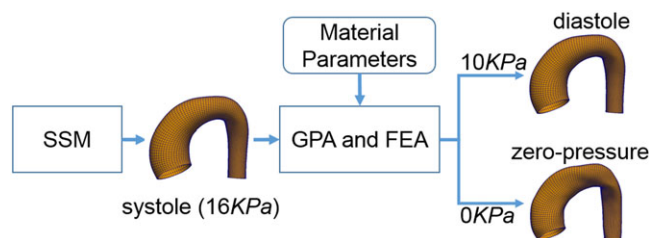
Recently, machine learning (ML) techniques, particularly deep learning<sup>22-24</sup> using deep neural networks (DNNs), have garnered enormous attention in the field of artificial intelligence, and have led to revolutionary breakthroughs in many applications,<sup>22,25-31</sup> which reach and in some cases, surpass human performance.<sup>25-28</sup> Since DNNs are capable of modeling complex and nonlinear relationship between variables, they could offer an alternative solution to obtain zero-pressure geometries. By designing and training a neural network on a large dataset generated by FEA-based methods, it could automatically produce the required output (ie, a zero-pressure geometry) directly from necessary inputs, without the need for time-consuming FE simulations, thereby acting as an ML surrogate of the FEA-based methods.

In this study, we developed, to our best knowledge, the first ML approach to estimate zero-pressure geometries, as a fast surrogate of FEA-based inverse methods. An ML model was developed, trained, and tested on a dataset of thoracic aorta shapes from 3125 virtual patients, which were generated using a FEA-based method. Given an input pair of aorta shapes at different pressure levels, the trained ML model can output the zero-pressure shape within 1 second, consistent with and magnitudes faster than the FEA-based method.

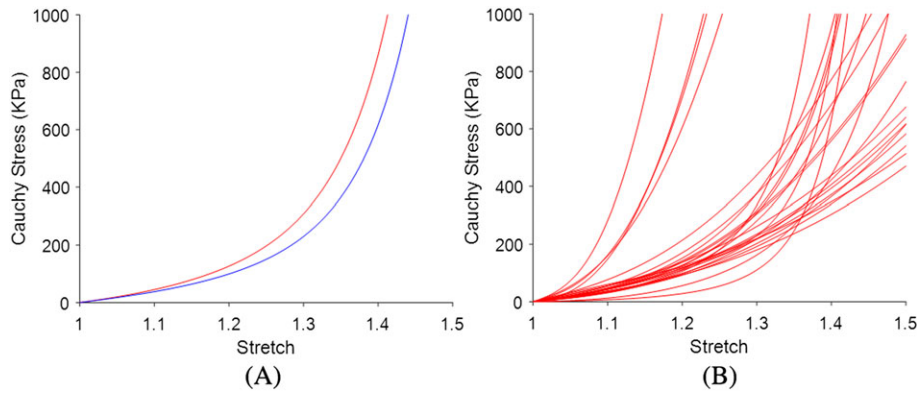
## 2 | METHODS

### 2.1 | FE simulation dataset of virtual patient geometries

We have constructed a dataset of 125 thoracic aorta shapes in a previous study,<sup>20</sup> which were generated from a statistical shape model (SSM)<sup>32,33</sup> built upon real patient geometries at the systolic phase (blood pressure of 16 KPa). Given a shape at the systolic phase and a set of material parameters, the generalized prestressing algorithm (GPA)<sup>13</sup> was used to obtain the prestressed state of the aorta at the systolic phase, which was then depressurized in FEA to obtain 2 corresponding geometries at the diastolic phase (blood pressure of 10 KPa) and the zero-pressure state. Each shape is represented by a solid mesh with 10 000 nodes and 4950 hexahedron elements, as shown in Figure 1, which is topologically equivalent to a flat rectangular mesh after cutting along the longitudinal direction. The mechanical behavior of the aorta tissue was modeled by a fiber reinforced hyperelastic constitutive model.<sup>34</sup> Twenty-five sets of material parameters from our previous mechanical testing experiments using tissue samples from 25 patients<sup>4</sup> were assigned to each of the 125 shapes at the systolic phase. The FE simulations were done by using Abaqus/Standard 6.14 (Simulia, RI): Details are provided in the Appendix. The resulting dataset of thoracic aorta shapes, at the systolic phase, diastolic phase, and zero-pressure state from 3125 (125 shapes from SSM  $\times$  25 sets of material parameters) virtual patients, was used to train and test the ML model (Section 2.2). Figure 1 shows the FEA-based method to obtain the zero-pressure geometry given 2 inputs: (1) a set of material parameters and (2) a geometry at a known blood pressure level. Figure 2 shows the stress-stretch curves of the 25 sets of material parameters in equibiaxial tensile tests simulated in MATLAB.



**FIGURE 1** The process to generate the aorta shapes of a virtual patient at the systolic phase (16 KPa), diastolic phase (10 KPa), and zero-pressure state (0 KPa). FEA, finite element analysis; GPA, generalized prestressing algorithm; SSM, statistical shape model



**FIGURE 2** Simulated equibiaxial tensile tests in circumferential and longitudinal directions were used to show (A) 2 stress-stretch curves of a set of material parameters. (B) Stress-stretch curves of the 25 sets of material parameters, which were obtained in the circumferential direction

## 2.2 | ML model

The ML model is a feedforward fully connected neural network with 4 hidden layers, and it is divided into 3 modules: shape encoding, nonlinear mapping, and shape decoding, as shown in Figure 3. Given a pair of input shapes at the systolic and diastolic phases, the ML model will output the zero-pressure shape in 3 steps: (1) encode each of the input shapes as a set of scalar values, ie, shape code; (2) transform the shape code of the input shapes to the shape code of the output shape using a nonlinear mapping; and (3) decode the shape code of the output shape into the zero-pressure shape. Note that on the basis of the previous studies,<sup>35-37</sup> the material parameters of the aorta wall tissue<sup>34</sup> can be identified from the aorta shapes at 2 cardiac phases with known blood pressure levels (eg, at systole and diastole), which implies that material property information is already embedded in the 2 input shapes. Therefore, the ML model does not require material parameters as input. The ML model is described in detail in the following sections.

### 2.2.1 | Shape encoding and decoding

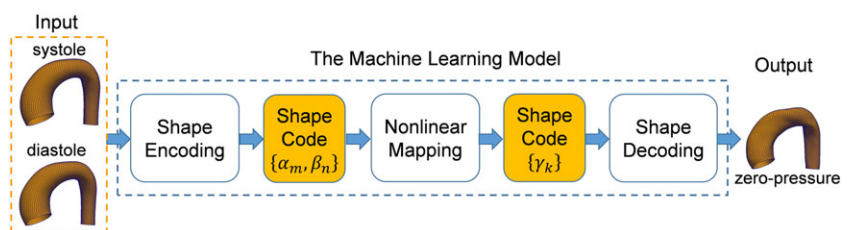
Shape encoding is the process of using a small number of scalar values to represent a shape, and shape decoding is the process of recovering the shape from its shape code. There are several methods available for shape encoding and decoding, including principal component analysis (PCA), independent component analysis, and sparse coding and dictionary learning.<sup>38</sup> Principal component analysis is widely used in statistical shape analysis<sup>33</sup> and was chosen as the shape encoding method in this study. By using PCA, a shape  $X$  can be represented by

$$X \cong \bar{X} + \sum_{i=1}^{l_{\max}} c_i \sqrt{\lambda_i} V_i, \quad (1)$$

where  $\bar{X}$  is the mean shape and  $\{V_i\}$  and  $\{\lambda_i\}$  are the eigenvectors (ie, modes of shape variations) and eigenvalues of the covariance matrix, respectively. In this study, the average PCA reconstruction error<sup>39</sup> is less than 1%. Given Equation 1 and the fact that eigenvectors are orthogonal to each other, the shape code  $\{c_i\}$  of  $X$  can be obtained by

$$c_i = V_i^T (X - \bar{X}) / \sqrt{\lambda_i}, \quad (2)$$

where  $V_i^T$  is the transpose of the column vector  $V_i$ .



**FIGURE 3** The overall structure of the machine learning model

Equations 1 and 2 can be conceptually represented as a bidirectional neural network with only one layer and linear units, as shown in Figure 4. Given the input positions of the 10 000 nodes, it outputs the shape code; and given the shape code as the input, it outputs the node positions. The weights of this neural network are determined by the eigenvectors and eigenvalues from PCA.

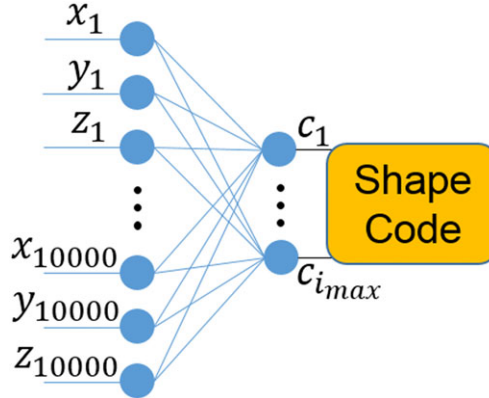
In the ML model (Figure 3), Equation 2 (ie, shape encoding) corresponds to the input layer and the first hidden layer, processing 2 input shapes to obtain 2 sets of shape code  $\{c_i\}$ , which were renamed as  $\{\alpha_m\}$  for the input shape at the systolic phase and  $\{\beta_n\}$  for the input shape at the diastolic phase. Equation 1 (ie, shape decoding) corresponds to the output layer of the ML model, recovering the zero-pressure shape from its shape code denoted as  $\{\gamma_k\}$  (ie, renaming  $\{c_i\}$  in Equation 1 as  $\{\gamma_k\}$ ), which is the output from the nonlinear mapping module. The subscripts  $i$ ,  $m$ ,  $n$ , and  $k$  are in the range of 1 to 64 in this application.

### 2.2.2 | Nonlinear mapping

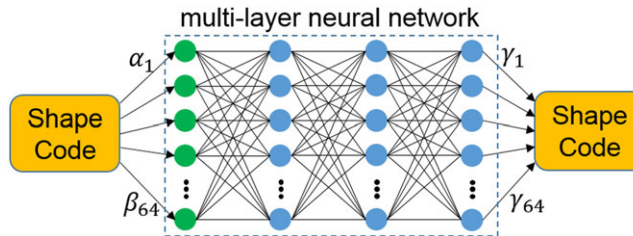
The nonlinear mapping module corresponds to the second, third, and fourth hidden layers of the ML model, mapping the shape code of the 2 input shapes to the shape code of the output zero-pressure shape, ie, establishing a set of nonlinear functions  $\gamma_k = f(\alpha_1, \dots, \alpha_M, \beta_1, \dots, \beta_N)$  for every  $k$ , where  $M = 64$  and  $N = 64$  in this application. As shown in Figure 5, this module is also a neural network: The inputs are  $\{\alpha_1, \dots, \alpha_{64}, \beta_1, \dots, \beta_{64}\}$ . There are 2 hidden layers. The output layer has 64 units, corresponding to  $\{\gamma_1, \dots, \gamma_{64}\}$ . Each hidden layer has the same number of Softplus<sup>40</sup> units. According to the universal approximation theorem and its variants,<sup>41-45</sup> 2-hidden layers should be enough for the nonlinear mapping module to convert the input shape code to the output shape code.

### 2.3 | Implementation and training of the ML model

The ML model was implemented using TensorFlow<sup>46</sup> and MATLAB. An NVidia GTX 1080 Ti graphics processing unit (GPU) was used for training. Given an input pair of shapes, the ML model can output the zero-pressure shape within 1 second (excluding initialization and file input and output time) on a PC with 3.4 GHz quad core CPU and 32 GB RAM.



**FIGURE 4** The neural network for shape encoding and decoding.  $(x_i, y_i, z_i)$  is equal to the position of node- $i$  on shape  $X$  minus the position of node- $i$  on the mean shape  $\bar{X}$ . The link (ie, weight) between  $c_i$  and  $x_i$  is the first component of  $V_i/\sqrt{\lambda_i}$ ; the link between  $c_i$  and  $z_{10\ 000}$  is the last component of  $V_i/\sqrt{\lambda_i}$ ; and so on



**FIGURE 5** The neural network for mapping the shape code  $\{\alpha_m, \beta_n\}$  of the input shape to the shape code  $\{\gamma_k\}$  of the output shape. The green dots represent the units of the input layer, and the blue dots represent the units of the hidden layers and the output layer of the neural network

The modules of the ML model were trained separately through supervised and unsupervised learning methods. The parameters of shape encoding and decoding modules were obtained by using PCA (Equations 1 and 2), which is unsupervised learning. After the shape encoding and decoding modules were trained, the parameters of the nonlinear mapping module were obtained through supervised learning by using a mean-square loss function and the AdaMax optimization algorithm.<sup>47</sup> For a detailed explanation of the layers, units, and related theories, we refer the reader to the book by Goodfellow et al.<sup>41</sup>

## 2.4 | Evaluation of the ML model

The performance of the ML model was evaluated through cross-validation to test whether the trained model could recover the zero-pressure geometries. We define a shape tuple as a set of 3 shapes at the systolic phase, diastolic phase, and zero-pressure state from the same virtual patient. Let  $\rho\%$  denote a percentage and  $0 < \rho < 100$ . In each round of the cross-validation,  $\rho\%$  of the shape tuples were randomly selected as the training data to determine the parameters of the ML model, and the remaining  $(100 - \rho)\%$  of the data were used as the testing data to obtain performance measures. Two performance measures were used to evaluate the accuracy of the estimated zero-pressure shapes: mean absolute error (MAE) and normalized mean absolute error (NMAE). The process was repeated 100 times, and the mean of each performance measure was calculated. For each zero-pressure shape, MAE is defined by

$$\text{MAE} = \frac{1}{10\,000} \sum_{i=1}^{10\,000} \|X_i - \tilde{X}_i\|, \quad (3)$$

where  $X_i$  is the 3D position vector of node- $i$  of the “true” zero-pressure shape from FEA and  $\tilde{X}_i$  is the corresponding position vector estimated by the ML model.  $\|\cdot\|$  is the Euclidean norm. For each zero-pressure shape, NMAE is defined by

$$\text{NMAE} = \frac{\text{MAE}}{\max\{2R\}} \times 100\%, \quad (4)$$

where  $\max\{2R\}$  is the maximum diameter of the “true” zero-pressure shape. Since the mesh of each shape is equivalent to a grid (Figure 1), the surface consists of closed circumferential curves, ie, a set of rings along the centerline. The equivalent radius,  $R$ , of a circumferential curve is defined as its length divided by  $2\pi$ . Then, the maximum diameter,  $\max\{2R\}$ , can be obtained.

## 3 | RESULTS

To evaluate the ML model, 2 numerical experiments were performed. In the first experiment, the number of units per hidden layer of the nonlinear mapping module was varied from 64 to 1024, and cross-validation with a fixed  $\rho\% = 90\%$  was conducted. The results are reported in Table 1, which shows that the performance of the ML model increases when the number of units in the nonlinear mapping module increases. In the second experiment, the number of units per hidden layer of the nonlinear module was fixed to 512, and the percentage  $\rho\%$  in cross-validation was varied from 50% to 90%. The results are reported in Table 2, which shows that the performance of the ML model increases when more data are available for training.

To visualize the results, 6 representative examples are shown in Figure 6, and the corresponding accuracy measures are reported in Table 3 in which each example is identified by the row index in Figure 6 (eg, example- $k$  is in the  $k$ th row in Figure 6). The shapes at the systolic phase are chosen such that shape code  $\alpha_m = -1$  or 0 or 1 for  $m \leq 3$  and  $\alpha_m \approx 0$  for  $m > 3$ .

**TABLE 1** Results from the first numerical experiment

Number of units	64	128	256	512	1024
Avg MAE, mm	1.3955	0.8946	0.7086	0.6045	0.5336
Avg NMAE, %	3.8473	2.4569	1.9426	1.6587	1.4627

Abbreviations: MAE, mean absolute error; NMAE, normalized MAE; Avg, average.



**TABLE 2** Results from the second numerical experiment

$\rho\%$	50	60	70	80	90
Avg MAE, mm	0.9023	0.7950	0.7107	0.6541	0.6045
Avg NMAE, %	2.4744	2.1802	1.9527	1.7961	1.6587

Abbreviations: MAE, mean absolute error; NMAE, normalized MAE; Avg, average.

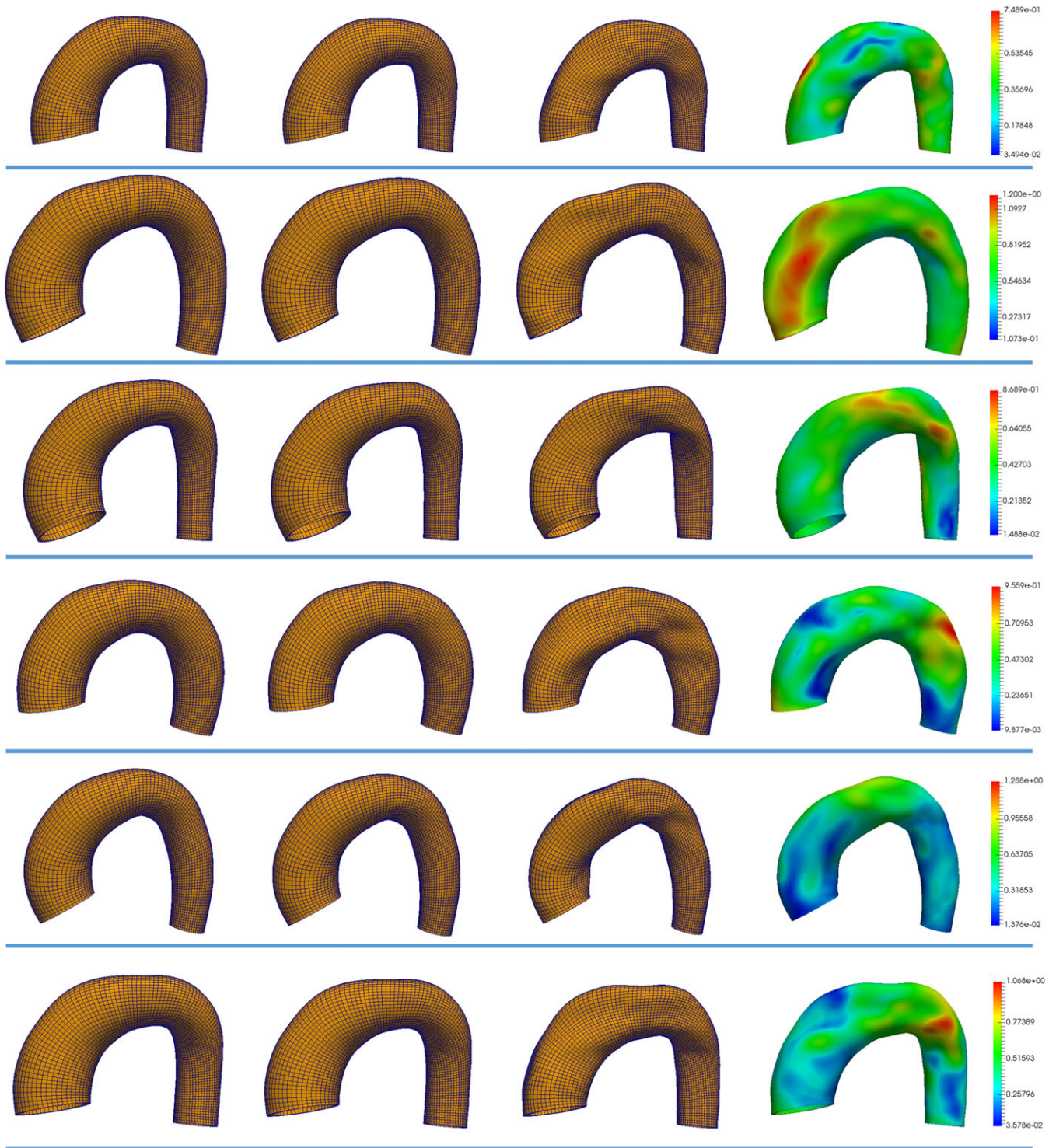
## 4 | DISCUSSION

In this study, we developed an ML model for zero-pressure geometry estimation, which was demonstrated on the application of the human thoracic aorta. To our knowledge, this is the first study in which an ML model was used as a surrogate of FEA-based inverse methods for recovering zero-pressure geometries. The results from the ML model are not only consistent with those of the FEA-based method (Figure 1) but also can be obtained magnitudes faster ( $<1$  s per input), excluding the time for training that can be done in advance. Given a shape at the systolic phase and a set of material parameters as the input, the time spent by the FEA-based method to obtain the zero-pressure shape varied from 3 to 50 minutes on the same PC. Furthermore, since the input to the ML model is only a pair of in vivo shapes, the material parameters associated with the pair of shapes must be identified before using the FEA-based method, which may take an additional 1 to 2 weeks using the FE-updating-based methods.<sup>35,36</sup> We note that the time for FEA may increase significantly if FE model complexity increases. Since efficient GPU implementation of DNNs is publicly available (eg, TensorFlow<sup>46</sup>) and GPU performance keeps increasing, DNNs can handle more complex geometries effectively by adding more layers and units. Thus, this ML approach has a clear advantage over FEA-based methods, making it more suitable for time-sensitive clinical applications.

We note that researchers have been developing GPU-based computational methods to accelerate FEA for biomedical applications.<sup>48–53</sup> However, regardless using CPU or GPU for FEA, numerical stability and convergence issues still exist.<sup>49</sup> Thus, for a single object study, a GPU-based FEA approach may be the best choice. However, if one needs to use FEA repeatedly for a huge number of similar objects, the proposed ML approach may provide a fast and robust solution.

Since the goal of this study was to evaluate the feasibility of using a neural network to estimate zero-pressure geometries, we made the following simplifications in FEA: (1) a uniform wall thickness at the systolic phase was assumed, (2) the branching vessels at the aortic arch were removed, and (3) residual stress was not considered. Currently, it is not possible to measure the wall thickness from computed tomography images because of the partial volume effect<sup>54</sup> and poor image contrast of the vessel tissue. However, it is possible to obtain the thickness distribution in vivo by using advanced magnetic resonance imaging.<sup>55</sup> It is also possible to include the branching vessels in the mesh model by stitching different mesh segments together. To incorporate residual stress, several methods in the literature<sup>19,56</sup> could be used to generate training data for the ML model. Also, by running GPA<sup>13</sup> and FEA with various blood pressure levels to generate training data, the ML model can be trained to work for a larger pressure range. Given a pair of shapes, it is difficult to obtain the real zero-pressure geometries (ie, the ground truth), and therefore, GPA and FEA were used to compute the zero-pressure geometries, which were considered as the “ground truth” for ML. Thus, any numerical errors from GPA and FEA will be inherited by the ML model.

The proposed ML model is not ready for clinical applications yet and needs to be further enhanced to handle a large range of shape variations. Since the SSM only covers some major shape variations,<sup>20</sup> a larger amount of patient data is needed to enhance the ML model. As suggested by a recent study,<sup>57</sup> DNN performance can be significantly improved by using more training data and fine tuning the structure, which will be applied in our future work. As more data are obtained, data heterogeneity will increase. Thus, to handle more complex shape distributions, the shape encoding and decoding algorithms may be changed. A good candidate will be multilinear PCA,<sup>58</sup> which is capable of handling more complex shape distributions than PCA and has been used to build a dynamic model of the left ventricle.<sup>59</sup> Another candidate will be sparse coding and dictionary learning,<sup>60,61</sup> which is able to handle complex shape distributions.<sup>62</sup> Neural networks can be used to implement/approximate sparse encoder/decoder by enforcing sparsity.<sup>63,64</sup> As the shape distribution becomes more complex, it may be challenging to maintain mesh correspondence. We previously developed a method for establishing mesh correspondence,<sup>20</sup> and this method can be enhanced in our future work. In this study, the ML model was not trained in an end-to-end fashion because the number of model parameters is much larger than the number of training shapes, which may cause overfitting issues. Thus, the ML model was divided into 3 modules that were trained separately.



**FIGURE 6** Six examples of estimated zero-pressure geometries. Each row shows the aorta shapes of a virtual patient (from left to right): the input shape at the systolic phase, the input shape at the diastolic phase, the “true” zero-pressure shape from finite element analysis, and the estimated zero-pressure shape from the machine learning model. The estimated shapes are color coded by the distance errors at each node. The unit is mm in the error legend

## 5 | CONCLUSION

In this study, an ML model was developed to directly estimate human thoracic aorta zero-pressure geometries from 2 in vivo geometries at 2 cardiac phases (ie, systole and diastole) that can be obtained from 3D in vivo image data. The

**TABLE 3** The accuracy of the estimated zero-pressure shapes in Figure 6

	Example 1	Example 2	Example 3	Example 4	Example 5	Example 6
$\{\alpha_1, \alpha_2, \alpha_3\}$	$\{1, 0, 0\}$	$\{-1, 0, 0\}$	$\{0, 1, 0\}$	$\{0, -1, 0\}$	$\{0, 0, 1\}$	$\{0, 0, -1\}$
MAE, mm	0.3674	0.6721	0.3665	0.3686	0.3867	0.3970
NMAE, %	1.0347	1.6891	0.9101	1.0111	1.0914	0.9846

Abbreviations: MAE, mean absolute error; NMAE, normalized MAE.

ML model is a feedforward fully connected neural network with 4 hidden layers, which is divided into 3 modules. Shape encoding and shape decoding modules were trained through the unsupervised method (ie, PCA). Nonlinear mapping module was trained through the supervised method. The ML model was evaluated through cross-validation on a dataset of aorta shapes from 3125 virtual patients, and the recovered zero-pressure geometries were consistent with those generated by the FEA-based method. The ML model can output the zero-pressure geometry within 1 second as opposed to FEA-based inverse methods that may take hours to days. This study clearly demonstrates the potential for ML techniques to estimate zero-pressure geometries of human organs as a fast surrogate of FEA-based inverse methods.

## ACKNOWLEDGEMENTS

Research for this project is funded in part by NIH, National Heart, Lung, and Blood Institute grant R01 HL104080 and R21 HL127570. Liang Liang is supported by an American Heart Association 16POST30210003.

## CONFLICT OF INTEREST

None declared.

## ORCID

Wei Sun  <http://orcid.org/0000-0002-8708-5128>

## REFERENCES

1. Sun W, Martin C, Pham T. Computational modeling of cardiac valve function and intervention. *Annu Rev Biomed Eng.* 2014;16(1):53-76.
2. Mao W, Caballero A, McKay R, Primiano C, Sun W. Fully-coupled fluid-structure interaction simulation of the aortic and mitral valves in a realistic 3D left ventricle model. *PLoS One.* 2017;12(9):e0184729.
3. Martin C, Sun W, Elefteriades J. Patient-specific finite element analysis of ascending aorta aneurysms. *Am J Physiol Heart Circ Physiol.* 2015;308:1306-1316.
4. Martin C, Sun W, Pham T, Elefteriades J. Predictive biomechanical analysis of ascending aortic aneurysm rupture potential. *Acta Biomater.* 2013;9(12):9392-9400.
5. Gasser TC. Biomechanical rupture risk assessment: a consistent and objective decision-making tool for abdominal aortic aneurysm patients. *AORTA.* 2016;4(2):42-60.
6. Raut SS, Chandra S, Shum J, Finol EA. The role of geometric and biomechanical factors in abdominal aortic aneurysm rupture risk assessment. *Ann Biomed Eng.* 2013;41(7):1459-1477.
7. Rengier F, Mehndiratta A, von Tengg-Kobligh H, et al. 3D printing based on imaging data: review of medical applications. *Int J Comput Assist Radiol Surg.* 2010;5(4):335-341.
8. Qian Z, Wang K, Liu S, et al. Quantitative prediction of paravalvular leak in transcatheter aortic valve replacement based on tissue-mimicking 3D printing. *J Am Coll Cardiol Img.* 2017;10(7):719-731.
9. Chia HN, Wu BM. Recent advances in 3D printing of biomaterials. *J Biol Eng.* 2015;9(1):4.
10. Raghavan ML, Ma B, Fillinger MF. Non-invasive determination of zero-pressure geometry of arterial aneurysms. *Ann Biomed Eng.* 2006;34(9):1414-1419.
11. Lu J, Zhou X, Raghavan ML. Computational method of inverse elastostatics for anisotropic hyperelastic solids. *Int J Numer Methods Eng.* 2007;69(6):1239-1261.
12. Gee MW, Förster C, Wall WA. A computational strategy for prestressing patient-specific biomechanical problems under finite deformation. *Int J Numer Methods Biomed Eng.* 2010;26(1):52-72.



13. Weisbecker H, Pierce DM, Holzapfel GA. A generalized prestressing algorithm for finite element simulations of preloaded geometries with application to the aorta. *Int J Numer Methods Biomed Eng*. 2014;30(9):857-872.
14. Bols J, Degroote J, Trachet B, Verheghe B, Segers P, Vierendeels J. A computational method to assess the in vivo stresses and unloaded configuration of patient-specific blood vessels. *J Comput Appl Math*. 2013;246:10-17.
15. Maas SA, Erdemir A, Halloran JP, Weiss JA. A general framework for application of prestrain to computational models of biological materials. *J Mech Behav Biomed Mater*. 2016;61:499-510.
16. Rausch MK, Genet M, Humphrey JD. An augmented iterative method for identifying a stress-free reference configuration in image-based biomechanical modeling. *J Biomech*. 2017;58:227-231.
17. Sellier M. An iterative method for the inverse elasto-static problem. *J Fluids Struct*. 2011;27(8):1461-1470.
18. Riveros F, Chandra S, Finol EA, Gasser TC, Rodriguez JF. A pull-back algorithm to determine the unloaded vascular geometry in anisotropic hyperelastic AAA passive mechanics. *Ann Biomed Eng*. 2013;41(4):694-708.
19. Genet M, Rausch MK, Lee LC, et al. Heterogeneous growth-induced prestrain in the heart. *J Biomech*. 2015;48(10):2080-2089.
20. Liang L, Liu M, Martin C, Elefteriades JA, Sun W. A machine learning approach to investigate the relationship between shape features and numerically predicted risk of ascending aortic aneurysm. *Biomech Model Mechanobiol*. 2017;1-15.
21. Votta E, Presicce M, Della Corte A, et al. A novel approach to the quantification of aortic root in vivo structural mechanics. *Int J Numer Methods Biomed Eng*. 2017;33(9):e2849-n/a.
22. LeCun Y, Bengio Y, Hinton GE. Deep learning. *Nature*. 2015;521(7553):436-444.
23. Litjens G, Kooi T, Bejnordi BE, et al. A survey on deep learning in medical image analysis. *Med Image Anal*. 2017;42:60-88. arXiv:1702.05747.
24. Shen D, Wu G, Suk H-I. Deep learning in medical image analysis. *Annu Rev Biomed Eng*. 2017;19(1):221-248.
25. He K, Zhang X, Ren S, Sun J. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. *IEEE International Conference on Computer Vision*; 2015.
26. Kokkinos I. Pushing the boundaries of boundary detection using deep learning. *Int. Conf. On learning Representations*; 2016.
27. Taigman Y, Yang M, Ranzato MA, Wolf L. DeepFace: closing the gap to human-level performance in face verification. *IEEE Conference on Computer Vision and Pattern Recognition*; 2014.
28. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*; 2016.
29. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Neural Information Processing Systems*; 2012.
30. Wu Y, Schuster M, Chen Z, et. al., Google's neural machine translation system: bridging the gap between human and machine translation. *Computing Research Repository Abs/160908144*; 2016.
31. Hannun A, Case C, Casper J, et. al., Deep speech: scaling up end-to-end speech recognition. *Computing Research Repository Abs/14125567*; 2014.
32. Cootes TF, Taylor CJ, Cooper DH, Graham J. Active shape models—their training and application. *Comput Vis Image Underst*. 1995;61(1):38-59.
33. Heimann T, Meinzer H-P. Statistical shape models for 3D medical image segmentation: a review. *Med Image Anal*. 2009;13(4):543-563.
34. Gasser TC, Ogden RW, Holzapfel GA. Hyperelastic modelling of arterial layers with distributed collagen fibre orientations. *J R Soc Interface*. 2006;3(6):15-35.
35. Wittek A, Karatolios K, Bihari P, et al. In vivo determination of elastic properties of the human aorta based on 4D ultrasound data. *J Mech Behav Biomed Mater*. 2013;27:167-183.
36. Wittek A, Derwich W, Karatolios K, et al. A finite element updating approach for identification of the anisotropic hyperelastic properties of normal and diseased aortic walls from 4D ultrasound strain imaging. *J Mech Behav Biomed Mater*. 2016;58:122-138.
37. Liu M, Liang L, Sun W. A new inverse method for estimation of in vivo mechanical properties of the aortic wall. *J Mech Behav Biomed Mater*. 2017;72:148-158.
38. Zhang S, Zhan Y, Metaxas DN. Deformable segmentation via sparse representation and dictionary learning. *Med Image Anal*. 2012;16(7):1385-1396.
39. Devijver PA. *Pattern Recognition: A Statistical Approach*. London, GB: Prentice-Hall; 1982.
40. Dugas C, Bengio Y. Fran, #231, o. B, #233, lisle, C. Nadeau, Ren, #233, Garcia, *Incorporating Second-Order Functional Knowledge for Better Option Pricing*. *Proceedings of the 13th International Conference on Neural Information Processing Systems*. Denver, CO: MIT Press; 2000:451-457.
41. Goodfellow I, Bengio Y, Courville A. *Deep Learning*. Cambridge, MA: The MIT Press; 2016.
42. Cybenko G. Approximation by superpositions of a sigmoidal function. *Math Control Signals Syst*. 1989;2(4):303-314.

43. Sonoda S, Murata N. Neural network with unbounded activation functions is universal approximator. *Appl Comput Harmon Anal*. 2017;43(2):233-268.
44. Shaham U, Cloninger A, Coifman RR. Provable approximation properties for deep neural networks. *Appl Comput Harmon Anal*. 2016;44(3):537-557.
45. Hornik K. Approximation capabilities of multilayer feedforward networks. *Neural Netw*. 1991;4(2):251-257.
46. Abadi M, Agarwal A, Barham P, et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems. Retrieved from <http://tensorflow.org/>; 2015.
47. Kingma DP, Ba J. Adam: a method for stochastic optimization, the 3rd international conference for learning representations. 2015.
48. Štrbac V, Pierce DM, Rodriguez-Vila B, Vander Sloten J, Famaey N. Rupture risk in abdominal aortic aneurysms: a realistic assessment of the explicit GPU approach. *J Biomech*. 2017;56(Supplement C):1-9.
49. Štrbac V, Pierce DM, Vander Sloten J, Famaey N. GPGPU-based explicit finite element computations for applications in biomechanics: the performance of material models, element technologies, and hardware generations. *Comput Methods Biomech Biomed Engin*. 2017;20(16):1643-1657.
50. Comas O, Taylor ZA, Allard J, Ourselin S, Cotin S, Passenger J. Efficient nonlinear FEM for soft tissue modelling and its GPU implementation within the open source framework SOFA. In: Bello F, Edwards PJE, eds. *Biomedical Simulation: 4th International Symposium, ISBMS 2008, London, UK, July 7-8, 2008 Proceedings*. Berlin Heidelberg, Berlin, Heidelberg: Springer; 2008:28-39.
51. Taylor ZA, Comas O, Cheng M, et al. On modelling of anisotropic viscoelasticity for soft tissue simulation: numerical solution and GPU execution. *Med Image Anal*. 2009;13(2):234-244.
52. Joldes GR, Wittek A, Miller K. Real-time nonlinear finite element computations on GPU—application to neurosurgical simulation. *Comput Methods Appl Mech Eng*. 2010;199(49):3305-3314.
53. Štrbac V, Pierce DM, Sloten JV, Famaey N. GPU-based fast finite element solution for nonlinear anisotropic material behavior and comparison of integration strategies. In: Joldes GR, Doyle B, Wittek A, Nielsen PMF, Miller K, eds. *Computational Biomechanics for Medicine: Imaging, Modeling and Computing*. Cham: Springer International Publishing; 2016:97-105.
54. Barrett JF, Keat N. Artifacts in CT: recognition and avoidance. *Radiographics*. 2004;24(6):1679-1691.
55. Dieleman N, van der Kolk AG, Zwanenburg JJM, et al. Imaging intracranial vessel wall pathology with magnetic resonance imaging. *Circulation*. 2014;130(2):192-201.
56. Pierce DM, Fastl TE, Rodriguez-Vila B, et al. A method for incorporating three-dimensional residual stretches/stresses into patient-specific finite element simulations of arteries. *J Mech Behav Biomed Mater*. 2015;47:147-164.
57. Sun C, Shrivastava A, Singh S, Gupta A. Revisiting unreasonable effectiveness of data in deep learning era. arXiv:1707.02968; 2017.
58. Vasilescu MAO, Terzopoulos D. Multilinear subspace analysis of image ensembles. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings. 2003; 2: pp. II-93-9.
59. Zhu Y, Papademetris X, Sinusas AJ, Duncan JS. Segmentation of the left ventricle from cardiac MR images using a subject-specific dynamical model. *IEEE Trans Med Imaging*. 2010;29(3):669-687.
60. Mairal J, Bach F, Ponce J, Sapiro G. Online learning for matrix factorization and sparse coding. *J Mach Learn Res*. 2010;11:19-60.
61. Aharon M, Elad M, Bruckstein A. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans Signal Process*. 2006;54(11):4311-4322.
62. Zhang S, Zhan Y, Dewan M, Huang J, Metaxas DN, Zhou XS. Towards robust and effective shape modeling: sparse shape composition. *Med Image Anal*. 2012;16(1):265-277.
63. Gregor K, LeCun Y. *Learning Fast Approximations of Sparse Coding, Proceedings of the 27th International Conference on International Conference on Machine Learning*. Haifa, Israel: Omnipress; 2010:399-406.
64. Rangamani A, Mukherjee A, Basu A, Ganapathy T, Arora A, Chin S, Tran TD. Sparse coding and autoencoders. arXiv:1708.03735; 2017.

**How to cite this article:** Liang L, Liu M, Martin C, Sun W. A machine learning approach as a surrogate of finite element analysis-based inverse method to estimate the zero-pressure geometry of human thoracic aorta. *Int J Numer Meth Biomed Engng*. 2018;e3103. <https://doi.org/10.1002/cnm.3103>

## APPENDIX A

The details about the finite element simulation dataset of the aorta shapes are described as follows.

### Image data

Cardiac computed tomography images of 25 thoracic aortic aneurysm patients, who underwent elective repair between the years<sup>3</sup> 2008 and 2010, were obtained from Yale New Haven Hospital. The images were deidentified, and clinical Institutional Review Board approval to analyze these data was obtained. The resolution of the images is  $0.7 \times 0.7 \times 2.5$  mm, and the field of view covers the thoracic and abdominal aorta. For each patient, the 3D aorta surface was semiautomatically reconstructed from the image data, and then it was trimmed at the ascending aorta just distal to the sinotubular junction on the proximal end and at the descending aorta on the distal end. The branch vessels at the arch were removed. The resulting surface was remeshed<sup>20</sup> to have 5000 nodes and 4950 elements, which established mesh correspondence between different patients. Each aorta mesh is topologically equivalent to a flat rectangular mesh. The solid mesh of each patient was obtained by offsetting the surface mesh by 1.5 mm (ie, thickness), which is based on our previous work.<sup>20</sup>

### Statistical shape modeling and virtual aorta geometry data

A statistical shape model (SSM)<sup>20,32,33</sup> was built from the patient data to generate a set of virtual aorta geometries at the systolic phase. Briefly, the patient aorta shapes were aligned to a common coordinate system, and then the principal component analysis algorithm<sup>32,33</sup> was invoked to build the SSM from these aligned shapes. A total number of 125 virtual aorta shapes were obtained by uniformly sampling the SSM parameters in the range of  $-2$  to  $2$ , ie, within 2 standard deviations of the mean shape. Further details are provided in our previous paper.<sup>20</sup>

### Finite element simulation and virtual patient data

Given one of the 125 shape at the systolic phase and a set of material parameters, finite element analysis with the generalized prestressing algorithm<sup>13</sup> was performed in Abaqus/Standard 6.14 (Simulia, RI) to obtain the prestressed state of the aorta at the systolic phase (blood pressure of 16 KPa), which was then depressurized to obtain the 2 geometries: one at the diastolic phase (blood pressure of 10 KPa) and the other one at the zero-pressure state. In the simulations, C3D8H solid elements were used, and pressures were applied uniformly to the inner surface of the aorta models, and the boundary nodes of the aorta models, ie, the proximal and distal ends of the model, were only allowed to move in the radial direction based on the local coordinate system.

A fiber reinforced hyperelastic material model based on the work of Gasser et al<sup>34</sup> was used to characterize the mechanical response. The strain energy function can be expressed as

$$\Psi = C_{10}(\bar{I}_1 - 3) + \frac{k_1}{2k_2} \sum_{i=1}^2 \left[ \exp \left\{ k_2 [\kappa \bar{I}_1 + (1 - 3\kappa) \bar{I}_{4i} - 1]^2 \right\} - 1 \right] + \frac{1}{D} \left[ \frac{J^2 - 1}{2} - \ln J \right], \quad (A1)$$

where  $C_{10}$  is a material parameter to describe the matrix material.  $k_1$  is a positive material parameter that has the same unit of stress, while  $k_2$  is a unitless material parameter. The strain invariant  $\bar{I}_1$  characterizes the matrix material; and the strain invariant  $\bar{I}_{4i}$  characterizes the fiber families.  $\bar{I}_{4i}$  is equal to squares of the stretches in the fiber directions. The material parameter  $\kappa$  describes the distribution of fiber orientation. A material parameter  $\theta$  defines the angle between mean local fiber direction and the circumferential axis of the local coordinate system.  $D$  is a constant that enforces material incompressibility and it is fixed to  $5 \times 10^{-4}$ .

Twenty-five sets of material parameters  $\{C_{10}, k_1, k_2, \kappa, \theta\}$  obtained by fitting the model (Equation (A1)) to the data from our previous experiment<sup>4</sup> were used in the finite element simulations. Then a dataset from 3125 (125 shapes from SSM  $\times$  25 sets of material parameters) virtual patients was obtained, consisting of 3 shapes per virtual patient at the systolic phase, diastolic phase, and zero-pressure state. We note that on the basis of the previous studies,<sup>35-37</sup> the material parameters of the constitutive model (Equation A1) can be identified from the aorta shapes at 2 cardiac phases with known blood pressure levels (eg, systole and diastole), which implies the 2 geometries contain material property information.