bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

A Deep Learning Approach to Estimate Collagenous Tissue Nonlinear Anisotropic Stress-Strain Responses from Microscopy Images

Liang Liang, Minliang Liu and Wei Sun

Tissue Mechanics Laboratory

The Wallace H. Coulter Department of Biomedical Engineering

Georgia Institute of Technology and Emory University, Atlanta, GA

Submitted to

Biorxiv

6/22/2017

For correspondence:

Wei Sun, Ph.D.

The Wallace H. Coulter Department of Biomedical Engineering
Georgia Institute of Technology and Emory University
Technology Enterprise Park, Room 206
387 Technology Circle, Atlanta, GA 30313-2412
Tel:(404) 385-1245; Email: wei.sun@bme.gatech.edu

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

ABSTRACT

Biological collagenous tissues comprised of networks of collagen fibers are suitable for a broad spectrum of medical applications owing to their attractive mechanical properties. In this study, we developed a noninvasive approach to estimate collagenous tissue elastic properties directly from microscopy images using Machine Learning (ML) techniques. Glutaraldehyde-treated bovine pericardium (GLBP) tissue, widely used in the fabrication of bioprosthetic heart valves and vascular patches, was chosen as a representative collagenous tissue. A Deep Learning model was designed and trained to process second harmonic generation (SHG) images of collagen networks in GLBP tissue samples, and directly predict the tissue elastic mechanical properties. The trained model is capable of identifying the overall tissue stiffness with a classification accuracy of 84%, and predicting the nonlinear anisotropic stress-strain curves with average regression errors of 0.021 and 0.031. Thus, this study demonstrates the feasibility and great potential of using the Deep Learning approach for fast and noninvasive assessment of collagenous tissue elastic properties from microstructural images.

Keywords: *Deep Learning, convolutional neural network, elastic property, collagenous tissue*

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

1. INTRODUCTION

Biological collagenous tissues are comprised of networks of collagen fibers embedded in a ground substance [1, 2], which provide pliability and strength important for many normal physiological functions. The attractive biological and mechanical properties [3] also make collagenous tissues, mostly derived from animals as xenografts, suitable for a broad spectrum of medical applications such as bioprosthetic heart valve (BHV) [4, 5], cardiovascular grafting/patch [6, 7], tendon [8] and hernia [9] repair. However, due to the heterogeneity and inherent variability of biological tissues, the mechanical properties of collagenous tissues obtained at different locations even within the same individual (regardless whether animal or human) may differ, and may impact tissue-derived device function.

Many studies [10-16] have shown that the microstructure of soft tissues, particularly the collagen fiber network structure, is the key determinant of the tissue elastic properties at the macroscopic level. Advanced microscopy imaging techniques, such as second harmonic generation (SHG) imaging, has enabled noninvasive visualization of soft tissue collagen networks at the microstructural level. The elastic properties of collagenous tissues are traditionally obtained through destructive mechanical testing of harvested tissue samples (Figure 1). Ideally, the nonlinear anisotropic elastic properties of collagenous tissues could be directly estimated from noninvasive images (e.g. SHG images) of the tissue microstructure, such that xenografts could be carefully selected based on their mechanical properties and optimal, more predictable, tissue-derived device function could be ensured.

Recently, Deep Learning [17], a branch of Machine Learning utilizing deep neural networks, has garnered enormous attention in the field of artificial intelligence. A special type of neural network, namely the convolutional neural network (CNN) [17-19], has become the state-

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

of-the-art approach for computer vision and image analysis applications (e.g. face recognition), reaching, and even surpassing, human performance in some cases [20-23]. CNN provides an end-to-end solution from input image to output target value by automatically extracting image features, thus eliminating the need for hand-crafted image features.

In this study, we developed, to our best knowledge, the first Deep Learning approach to estimate the elastic properties of collagenous tissues from SHG images (Figure 1). Glutaraldehyde-treated bovine pericardium (GLBP) tissue, widely used in the fabrication of BHVs [5] and vascular patches, was chosen as a representative collagenous tissue. A multi-layer CNN was designed and trained on a dataset of SHG images and corresponding mechanical testing results (i.e., equi-biaxial stress-strain curves). The trained CNN can automatically extract features from input SHG images of GLBP tissues and predict the nonlinear anisotropic elastic properties (Figure 1).

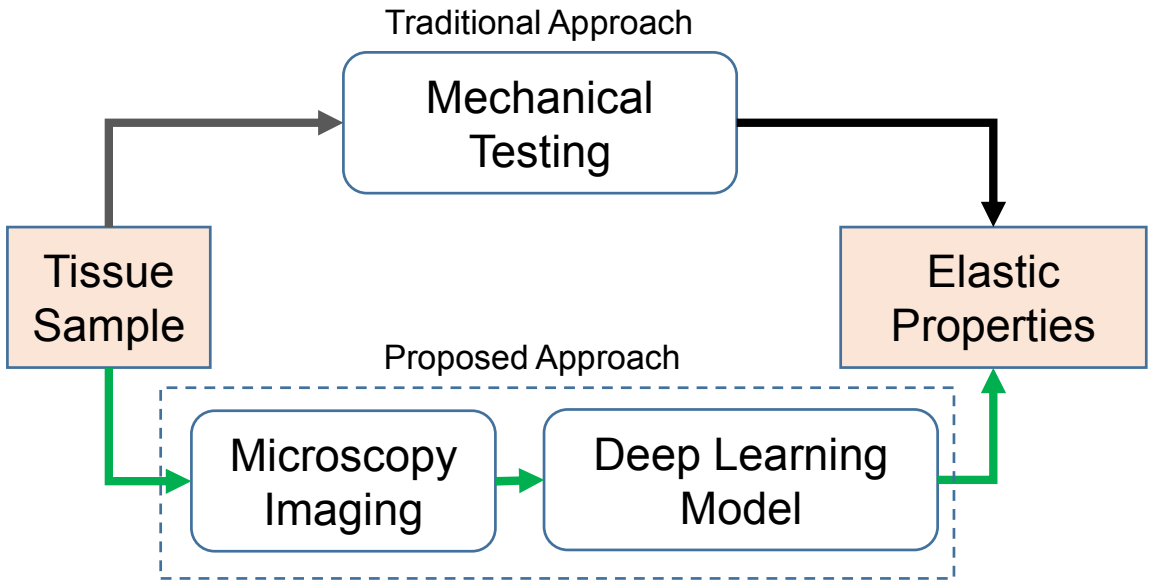


Figure 1. Two approaches to obtain the elastic properties of a tissue sample: 1) the traditional approach utilizing mechanical testing of a physical test sample and 2) noninvasive microscopy imaging coupled with a trained Deep Learning model.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2. METHODS

2.1 Tissue preparation and mechanical testing

The GLBP tissue samples used in this study were collected and mechanically tested through previous work by our group aimed at evaluating transcatheter heart valve biomaterials [24]. The tissue preparation and mechanical testing protocols are well documented in the published works [25-27]. Briefly, testing samples were cut into a 20×20 mm^2 square, and four graphite markers delimiting a square approximately 2×2 mm^2 in size were glued to the central region of each sample for optical strain measurements. Samples were then mounted in a trampoline fashion to a planar biaxial tester in aqueous 0.9% NaCl solution at 37 °C. A stress-controlled test protocol [25] was utilized to obtain the biaxial stress-strain response curves of each testing sample. In this study, 48 GLBP tissue samples were tested in total.

2.2 Tissue imaging

Upon completion of biaxial mechanical testing, the tissue samples were imaged using the SHG technique at the unloaded state. We utilized a Zeiss 710 NLO inverted confocal microscope (Carl Zeiss Microscopy, LLC, Thornwood, NY, USA), equipped with a mode-locked Ti:Sapphire Chameleon Ultra laser (Coherent Inc., Santa Clara, CA), a non-descanned detector (NDD), and a Plan-Apochromat 40x oil immersion objective. The laser was set to 800 nm and emission was filtered from 380–430 nm. Samples were kept hydrated with saline solution during imaging to prevent drying artifacts and covered with #1.5 coverslips. Samples were imaged inside the area delimited by the graphite markers, and 2D image slices were collected in the thickness direction from the smooth side of each sample. A 2D slice has 512×512 pixels to 1024×1024 pixels, and for each sample the number of slices was varied to cover the thickness. In total, we obtained 3D SHG images (size from 512×512×N to 1024×1024×N) of 48 tissue samples from different animal

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

subjects, and the corresponding mechanical testing data. Representative SHG images of a GLBP sample are shown in Figure 2, with a total of 18 slices (N=18) through the thickness. It is evident from Figure 2 that the image patterns change very slowly through the GLBP tissue thickness.

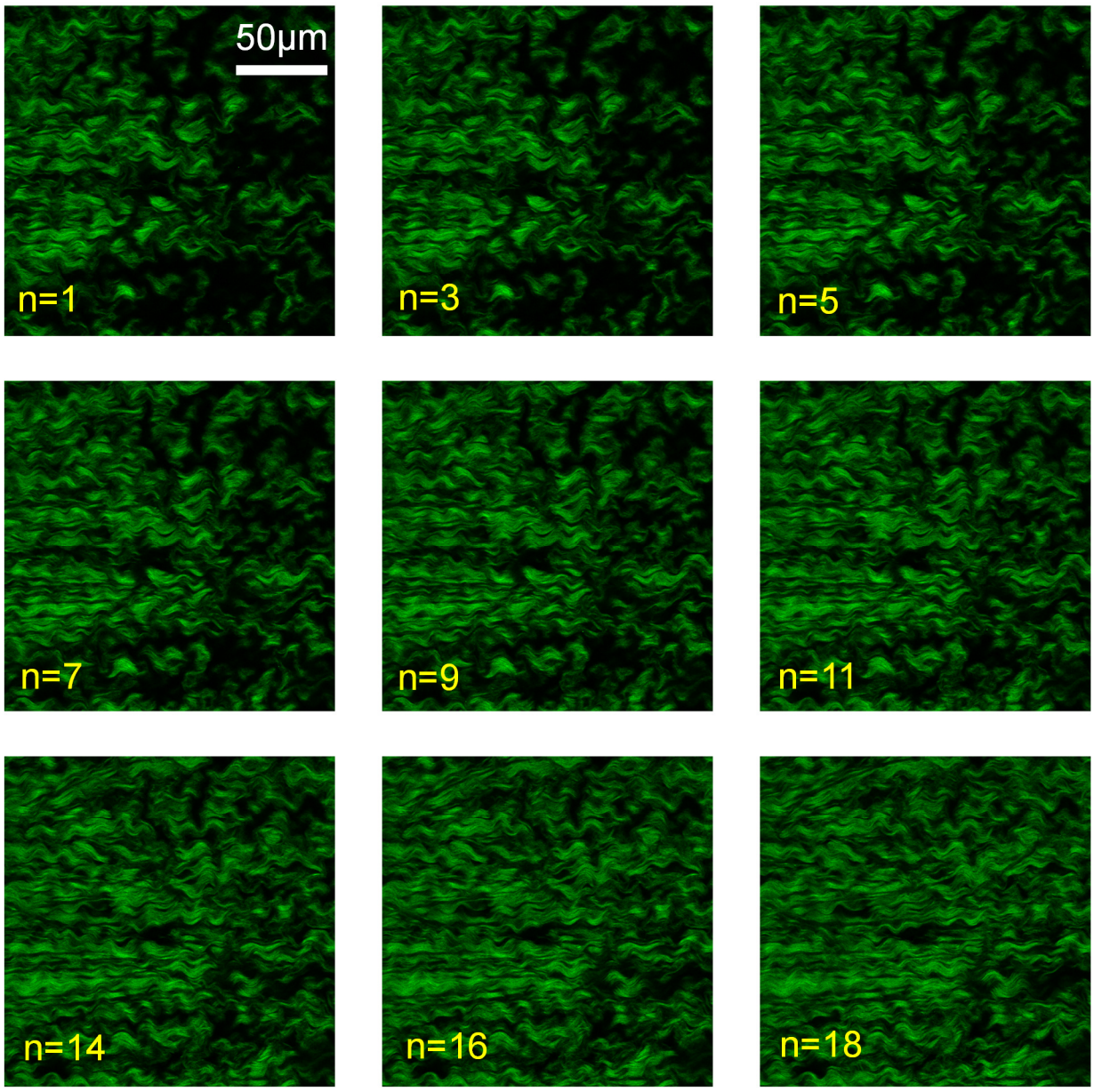


Figure 2. Representative SHG image slices of a tissue sample. n denotes the index of each slice.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2.3 PCA-based Parameterization of GLBP stress-strain curves

Two distinct stress-strain curves were obtained from the equi-biaxial mechanical testing (section 2.1) of each tissue sample (Figure 3a&b), due to the anisotropic mechanical behavior of the tissue : 1) strain E_{11} and stress S_{11} along the X_1-direction, and 2) strain E_{22} and stress S_{22} along the X_2-direction. Each stress-strain curve was uniformly sampled along the stress axis within the range of 10 to 630 *KPa*. The cutoff of 630*KPa* was chosen because different ranges of external stresses were applied to the tissue samples and 630*KPa* was the minimum peak stress value. For each tissue sample, the resampled strain values from the two curves were assembled as a vector of 126 numbers, Y. By using principle component analysis (PCA) [28, 29], the vector Y of a tissue sample can be decomposed as

$$Y \cong Y_{PCA} = \bar{Y} + \alpha_1 V_1 + \alpha_2 V_2 + \alpha_3 V_3 \quad (1)$$

where $\bar{Y}$ is the population mean, $\{V_i\}$ are the modes of variation, and $\{\alpha_i\}$ are the coefficients. Here, $\{\alpha_i\}$ can vary, while $\bar{Y}$ and $\{V_i\}$ are the same for all tissue samples. The first three modes of variation $\{V_1, V_2, V_3\}$ with $\{\alpha_1, \alpha_2, \alpha_3\}$ can describe 99% of the total variation of the stress-strain curves, which means each stress-strain curve can be almost perfectly reconstructed by using Eq.(1) as shown in Figure 3b. Furthermore, the reconstruction error was measured by the mean absolute error (MAE), given by

$$\text{MAE} = \frac{1}{L_2 - L_1 + 1} \sum_{j=L_1}^{L_2} |Y_{PCA}(j) - Y(j)| \quad (2)$$

where j is the index of a component in a vector; and if $L_1 = 1, L_2 = 63$, MAE is the error of the reconstructed S_{11}~E_{11} curve; and if $L_1 = 64, L_2 = 126$, MAE is the error of the reconstructed S_{22}~E_{22} curve.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

As shown in Figure 3c&d, a material is softer, i.e. more compliant, than the mean material if $\alpha_1 < 0$, and stiffer than the mean material if $\alpha_1 > 0$. Thus the sign of α_1 can be used to describe the overall tissue stiffness.

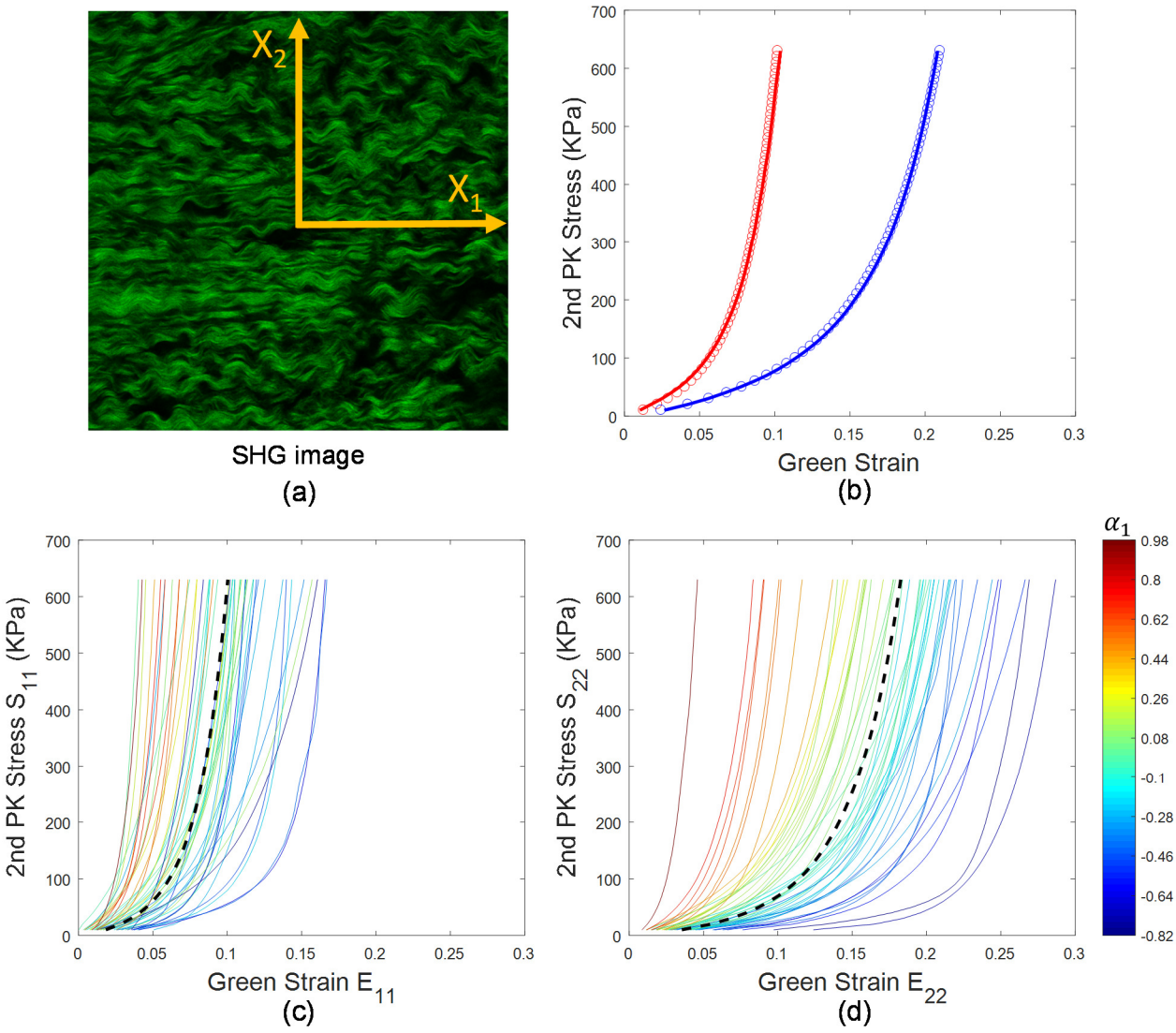


Figure 3. (a) The orientation definition of a tissue sample: X_1 direction and X_2 direction. (b) The open circles represent the stress-strain curves of a tissue sample from equi-biaxial mechanical testing experiments. The reconstructed stress-strain curves are shown by the red lines (S_{11}~E_{11}) and blue lines (S_{22}~E_{22}). (c)&(d) The stress-strain curves in the two directions of the 48 tissue samples color-coded by the corresponding α_1. The dashed lines are the mean curves, $\bar{Y}$.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2.4 Deep learning model

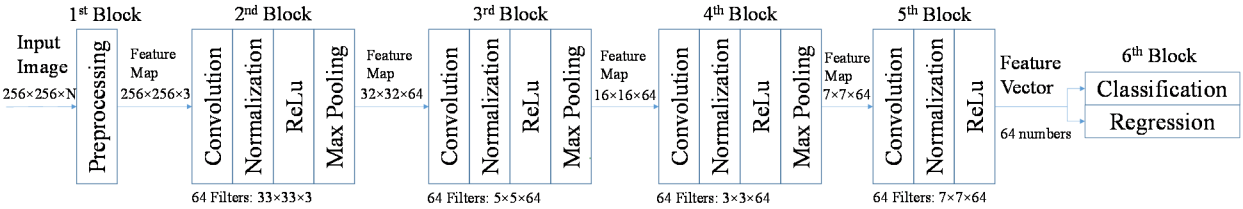


Figure 4. Architecture of the deep convolutional neural network used in this study.

As show in Figure 4, we designed a deep convolutional neural network (CNN) as the deep learning model, consisting of 6 blocks in a pipeline. The 1st block takes an input image of size 256×256×N pixels. The 6th block can be configured either as a classifier of the overall tissue stiffness (sign of α_1), or a regressor to predict the PCA parameters $\{\alpha_1, \alpha_2, \alpha_3\}$, which can be used to reconstruct the stress-strain curves by Eq.(1). The CNN (Figure 4) learns the relationship between the tissue SHG images and elastic properties from the training dataset, and then can infer the elastic properties from a new tissue image.

Usually, convolutional neural networks (CNNs) consist of many layers that are sequentially connected, e.g., output from the first layer is the input to the second layer. A layer performs a specific operation, such as convolution, normalization, or max pooling, and it has parameters either prescribed or to be learned from data. For a detailed explanation of these layers, we refer the reader to the reference papers [17, 18, 30, 31]. The network structure should be designed for specific applications, e.g., choosing the types and sizes of layers and determining their combinations. For our application, the designed CNN consisting of 6 blocks in a pipeline, where each block has one or more layers. Given an input 3D image of 256×256×N pixels, the 1st block with only one preprocessing layer, performs local contrast normalization and uniformly resamples the input 3D image into the first feature map of 256×256×3 pixels. The 1st block does not have any trainable/free parameters. The 2nd block contains a convolution layer with 64 filters

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

(a.k.a. kernels) of 33×33×3 pixels, a batch-normalization layer, a ReLu (rectifier linear unit) layer, and a max pooling layer; and the output from the 2nd block is a feature map of 32×32×64 pixels. The 3rd to the 5th blocks are very similar to the 2nd block, which output feature maps of 16×16×64, 7×7×64, and 1×1×64 pixels respectively. All of the max-pooling layers use a 2×2 pooling window. The 1st to 5th blocks can be considered image-feature extractors which output a feature vector of 64 numbers. The 6th block is used for classification with a softmax classifier, and regression with a linear model. The CNN was implemented by using MatConvnet [32], an open source MATLAB toolbox, and custom MATLAB functions; and it can process an input 3D image within 10 seconds on a PC with intel i7-4770 CPU and 32G RAM.

2.5 Learning of the deep convolutional neural network

The CNN (Figure 4) parameters were learned from the training data. To overcome the challenge of training the CNN with a small dataset [28] (i.e., 48 test samples, which is an acceptable sample size for material testing of biological tissues), the CNN was trained by combining: 1) unsupervised deep learning to determine the parameters in the 2nd to 5th blocks, 2) supervised learning to determine the parameters in the 6th blocks, and 3) data augmentation to generate more training data.

2.5.1 Unsupervised Deep Learning from the 2nd to 5th blocks

To determine the filter parameters of a convolution layer, generally we could use encoder-decoder based unsupervised learning strategies [33-36]. The input feature map to the convolution layer can be divided into small patches, where each patch has the same size as a filter (all filters in the same layer have the same size). Each patch can be converted to a vector, X, and the vectorized patches can be stacked together as the columns of a data matrix $\boldsymbol{X}$. The filters of the convolution layer can also be vectorized and stacked together as the columns of a filter matrix $\boldsymbol{A}$. Let $h(x)$

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

denote the ReLu function: $h(x) = x$ if $x > 0$, and $h(x) = 0$ if $x \leq 0$. The encoder performs convolution followed by ReLu to each patch, which outputs the code matrix $h(\boldsymbol{AX})$ close to the optimal (unknown yet) code matrix $\boldsymbol{Z}$. Given the optimal code matrix $\boldsymbol{Z}$, the decoder tries to recover the input patches $\boldsymbol{X}$ by using a linear combination of the atoms/columns in a dictionary/matrix $\boldsymbol{D}$, i.e, using $\boldsymbol{DZ}$ to approximate $\boldsymbol{X}$. Then the goal is to find the optimal variables $\{\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{Z}\}$ such that the encoding error and the decoding error are both minimized, which is to minimize the following objective function:

$$\mathcal{F} = \|h(\boldsymbol{A}'\boldsymbol{X}) - \boldsymbol{Z}\|^2 + \|\boldsymbol{X} - \boldsymbol{DZ}\|^2 + g(\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{Z}) \quad (3)$$

where $g(\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{Z})$ defines some constraints on the variables, and $\boldsymbol{A}'$ denotes the matrix-transpose of $\boldsymbol{A}$. The matrix norm $\|.\|$ is the Frobenius norm. Obviously, by using different constraints, we can obtain different solutions of $\{\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{Z}\}$. We proposed an algorithm with three steps to directly obtain a solution under the low rank constraint [37]:

Step-1: Perform low rank approximation (LRA) [37] on the patches $\boldsymbol{X}$, then a vectorized patch X can be approximated by

$$X \cong \sum_{m=1}^{M} z_m d_m = \boldsymbol{D}Z \,, \quad (4)$$

where $\boldsymbol{D} = [d_1, \ldots, d_M]$, and the vector d_m has the same size as X, and $M \leq \widetilde{M}$ which is the number of pixels in the patch X. d_m is the product of the m^{th} largest singular value, λ_m, and the corresponding left-singular vector obtained by LRA. $\boldsymbol{D}$ is the same for every single patch X. Also obtained by LRA, the code vector, $Z = [z_1, \ldots z_M]'$, is a column vector of scalars, which is different for different patches. The percentage error of approximation for the patches $\boldsymbol{X}$ is given by

$$Error = \frac{\sum_{n=M+1}^{\widetilde{M}} \lambda_n^2}{\sum_{m=1}^{\widetilde{M}} \lambda_m^2} \times 100\% \,. \quad (5)$$

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

If $M = \widetilde{M}$, then the error is zero. By controlling the number of retrained singular values and singular vectors, i.e., M, the approximation error and the computation cost (proportional to M) can be controlled. In this study, M is fixed to 32, and the error is less than 30%. The low rank approximation essentially obtains $\boldsymbol{D}$ and $\boldsymbol{Z}$ that minimize $\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{Z}\|^2$ under the low rank constraint. Since the singular vectors in $\boldsymbol{D}$ are orthogonal to each other, the code vector Z can be simply approximated by $\boldsymbol{D}'X$, i.e., $Z \cong \boldsymbol{D}'X$, which is obtained by multiplying $\boldsymbol{D}'$ to both sides of Eq.(4). After this step, the code matrix $\boldsymbol{Z}$ and dictionary $\boldsymbol{D}$ are determined.

Step-2: Define the filter matrix $\boldsymbol{A}$ by using the learned dictionary $\boldsymbol{D}$, given by

$$\boldsymbol{A} = [\boldsymbol{D}, -\boldsymbol{D}] = [d_1, \dots, d_M, -d_1, \dots, -d_M]. \quad (6)$$

Also, we define a new code vector $\tilde{Z}$ as

$$\tilde{Z} = [h(Z'), h(-Z')]' \ . \quad (7)$$

Then the objective function Eq.(3) is equivalent to

$$\mathcal{F} = \left\|h(\boldsymbol{A}'\boldsymbol{X}) - \widetilde{\boldsymbol{Z}}\right\|^2 + \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{Z}\|^2 + g(\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{Z}) \ , \quad (8)$$

where $\widetilde{\boldsymbol{Z}}$ is the stack of new code vectors, corresponding to $\boldsymbol{Z}$, and $\widetilde{\boldsymbol{Z}} \cong h(\boldsymbol{A}'\boldsymbol{X})$ because $Z \cong \boldsymbol{D}'X$. Then, a vectorized patch X can be encoded as a vector $\tilde{Z}$ by the encoder $h(\boldsymbol{A}'X)$. For example, $X \cong 2d_1 - 3d_2$, then the code vector is $[2, 0]$ if $\boldsymbol{A} = [d_1, d_2]$, and the code vector is $[2, 0, 0, 3]$ if $\boldsymbol{A} = [d_1, d_2 - d_1, -d_2]$, which clearly shows that the longer code vector preserves more information of X. The rationale of Eq.(6)&(7) is that the ReLu layer rejects any negative signal (i.e. code) output from the convolution layer, and therefore, nearly half of the signals will be lost in each block, harming the performance of the CNN. After this step, the filters of the convolution layer are determined.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Step-3: Perform feature map normalization. The output from the ReLu layer is a feature map serving as the input to the next layer. The size of the feature map is $K_1 \times K_2 \times K_3$ (i.e. height $\times$ width $\times$ channel). The values of the feature map at one spatial location can be assembled to a code vector Z of length K_3. By assembling all of the code vectors from the training dataset, a data matrix is obtained, and each row of this matrix is normalized by subtracting the mean and dividing by the standard deviation. The rows of the code matrix $\boldsymbol{Z}$ from a single input image will also be normalized in the same way by using the same values of mean and standard deviation. This normalization is essentially equivalent to batch-normalization [30] which has been shown to improve CNN accuracy. After this step, the parameters (i.e. mean and standard deviation values) of the normalization layer are determined.

2.5.2 Supervised learning in the 6th block

The 6th block can be configured either as a classifier or regressor. In the classification configuration, a softmax function is used to predict class membership based on the feature vector from the 5th block. Since it is a binary (soft vs. stiff) classification task, the softmax function reduces to a logistic function, given by

$$y = \frac{1}{1+exp\left(-\sum_{i=1}^{64} w_i x_i + b\right)} \tag{9}$$

where $\{w_1, \dots, w_{64}, b\}$ are the unknown scalar parameters and $[x_1, \dots, x_{64}]$ is the feature vector from the 5th block. Usually, a discrimination threshold (e.g. 0.5) is specified for the binary classification. If y is greater than or equal to the threshold, then the input is classified as stiff; and if y is smaller than the threshold, then it is classified as soft. With the labeled training data (i.e., image data with known mechanical properties), the 65 parameters in Eq.(9) can be determined through supervised learning using the cross-entropy loss function and the conjugate gradient optimization algorithm.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

In the regression configuration, a multiple output linear regressor predicts the values of $\{\alpha_1, \alpha_2, \alpha_3\}$ in Eq.(1) based on the feature vector from the 5th block, which is given by

$$\alpha_i = \sum_{j=1}^{64} w_{ij} x_j + b_i,\ i=1, 2, 3 \quad (10)$$

where $\{w_{ij}, b_i, i = 1,2,3, j = 1, \ldots,64\}$ are the unknown scalar parameters. With the labeled training data, the 195 parameters of this regressor can be learned by using the least squares regression algorithm. Once the parameters $\{\alpha_1, \alpha_2, \alpha_3\}$ are predicted by the regressor, the stress-strain curves can be reconstructed by using Eq.(1).

2.5.3 Data augmentation

Data augmentation methods are extensively used in Deep Learning applications [18, 38-40] to generate more training data. In this study, two data augmentation methods were used: image splitting and flipping (Figure 5). A 3D image of *N* slices can be split into patches using a sliding window with a stride of 128, and the size of each patch is 256×256×*N*. As a result of image splitting, 1678 patches were generated. Furthermore, each patch was flipped along the horizontal direction and/or vertical direction, which produced 6712 patches. The elastic properties corresponding to image patches from the same GLBP tissue sample, were assumed to be identical.

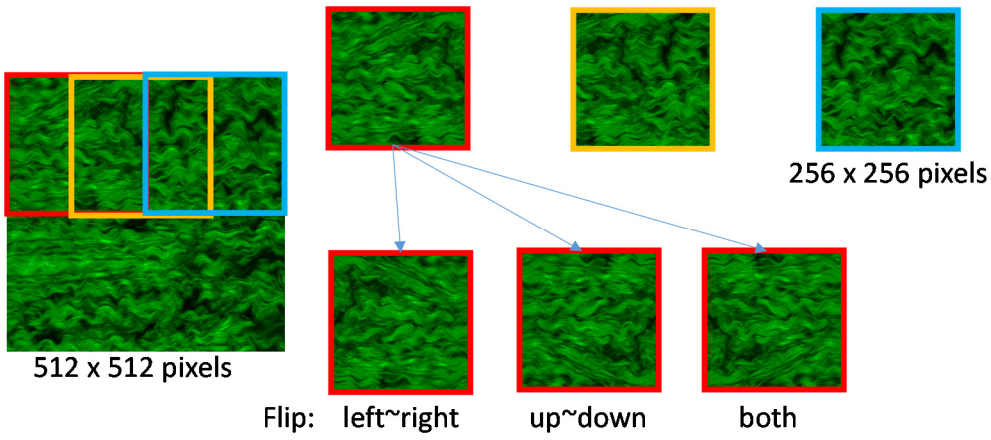


Figure.5 An example of data augmentation to generate image patches.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2.6 A comparative study of network structures

Given the relatively large size of the CNN compared to the dataset, the natural question arises whether reducing the number of layers or filters will significantly impact the performance. Given the huge design space, it would be impractical to evaluate all possible simplifications of the CNN structure. In this study, we chose to investigate two simplified CNNs for comparison, named CNN-s1 and CNN-s2 respectively. *CNN-s1*: in *Step-2* of unsupervised learning in section 4.5.1, the filter matrix $\boldsymbol{A}$ was simplified as $\boldsymbol{A} = \boldsymbol{D}$, which reduces the number of filters. *CNN-s2*: the ReLu and normalization layers were removed, and the filter matrix $\boldsymbol{A}$ was simplified as $\boldsymbol{A} = \boldsymbol{D}$. The structure of CNN-s2 is similar to that in [34].

3. RESULTS

3.1 Unsupervised deep learning

The learned filters of the CNN are visualized in Figure 6. The filters in the 2nd block (Fig. 6a) are local image feature detectors, resembling the local fiber network structures. The filters in the other blocks (Fig. 6b-d) are more abstract, essentially representing various combinations of the local structures at different length scales and locations.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

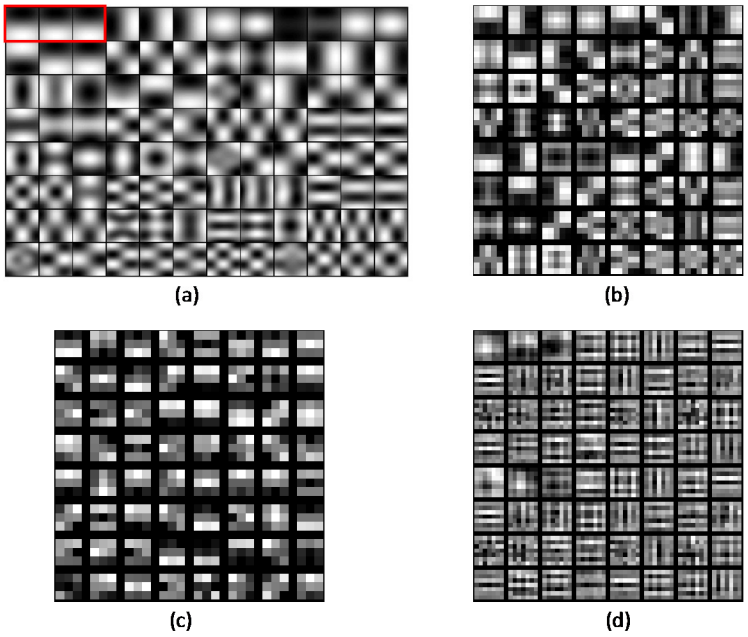


Figure 6. Examples of the learned filters. (a) The 32 filters in the convolution layer of the 2nd block, the 32 opposites of these filters are not shown. The red box contains one filter (size is 33×33×3). (b) One of the filters in the convolution layer of the 3rd block. (c) One of the filters in the convolution layer of the 4th block. (d) One of the filters in the convolution layer of the 5th block.

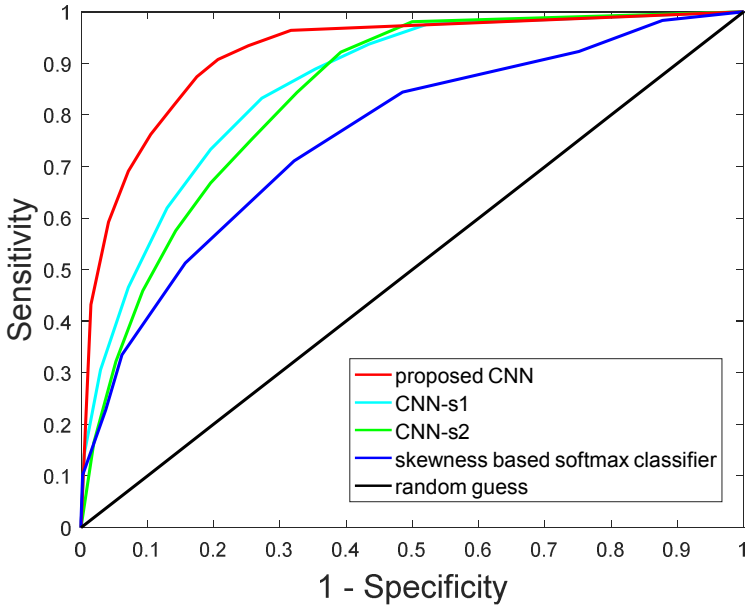
3.2 Classification

Classification performance was evaluated through ten-fold cross validation using the image patch data. In each round of cross validation, 90% of the image patches and corresponding overall stiffness values (i.e. sign of α_1) were randomly selected as the training data; and the remaining 10% of the data were used as the testing data to test whether the trained classifier can predict the sign of α_1, i.e., identify whether the tissue sample (corresponding to an image patch) is soft or stiff. The classification accuracy, defined as (TP+TN)/(TP+TN+FP+FN), the sensitivity, defined as TP/(TP+FN), and the specificity defined as TN/(TN+FP), were calculated to assess performance. Here, true positive (TP) is the number of stiff tissue patches correctly identified as stiff; false negative (FN) is the number of stiff tissue patches incorrectly identified as soft; true

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

298 negative (TN) is the number of soft tissue patches correctly identified as soft; and false positive
299 (FP) is the number of soft tissue patches incorrectly identified as stiff. In addition, AUC, defined
300 as the area under a receiver operating characteristic (ROC) curve, was calculated as a measure of
301 the overall classification performance. For comparison, a baseline softmax classifier using the
302 skewness of image histogram [41] as the only feature, was also trained and tested. Since the two
303 histograms of an image and its flipped version are the same, the flipped image patches were not
304 used in the classification experiment. Two simplified versions of the CNN, CNN-s1 and CNN-s2
305 with less filters and less layers (details in Method section), were also tested.

306 ROC curves, as shown in Figure 7, were obtained by varying the discrimination threshold
307 for each classifier. The performances of the proposed CNN, CNN-s1, CNN-s2, and the skewness-
308 based softmax classifier using 0.5 as the discrimination threshold for classification, are reported in
309 Table-1. The proposed CNN achieved the best performance, the skewness-based softmax classifier
310 had the worst performance, and the two simplified CNNs had moderate performance.



311

312 Figure 7. ROC curves of different classifiers

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Table-1: Classification Performance

Method	Accuracy	Sensitivity	Specificity	AUC
proposed CNN	84±2.5%	82±4.1%	86±3.6%	0.92
CNN-s1	78±2.8%	73±5.5%	80±3.7%	0.86
CNN-s2	75±3.5%	67±5.6%	80±4.9%	0.84
skewness based softmax classifier	71±3.2%	51±5.9%	84±3.9%	0.76

3.3 Regression

Regression performance was evaluated using a leave-one-out cross validation approach to test whether the trained regressor can predict the values of $\{\alpha_1, \alpha_2, \alpha_3\}$, which were used to reconstruct the stress-strain curve of each tissue sample by Eq.(1). In each round of the cross validation, the image patches and the stress-strain curves from one of the 48 tissue samples were used as the testing data to evaluate the accuracy of the regressor, and the remaining data were used as the training data to determine the parameters of the regressor. The predicted $\{\alpha_1, \alpha_2, \alpha_3\}$ values for each of the image patches from the test tissue sample were averaged to obtain the final $\{\alpha_1, \alpha_2, \alpha_3\}$ predictions for the whole tissue sample.

From the cross validation, the errors (Eq.(2)) in the predicted stress-strain curves were 0.021±0.015 and 0.031±0.029, compared to the actual S_{11}~E_{11} and S_{22}~E_{22} curves, respectively. Figure 8a shows an exemplary set of experimentally measured and predicted curves for one sample, and the error distribution across all of the samples is given in Figure 8b. The full set of predicted curves for all 48 samples are provided in the appendix.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

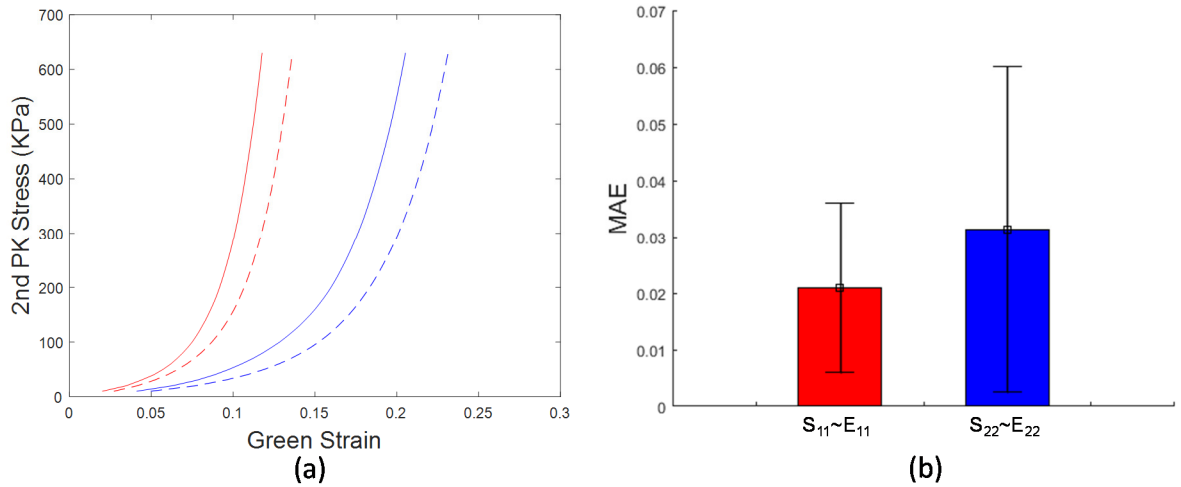


Figure 8. (a) Representative stress-strain curves predicted by the deep learning model shown as dashed lines, and the stress-strain curves obtained from mechanical testing shown as solid lines. S_{11}~E_{11} curves are shown in red. S_{22}~E_{22} curves are shown in blue. (b) The mean absolute error (MAE) distribution of all samples.

4. DISCUSSION

In this study, we developed a Deep Learning approach utilizing a deep CNN to estimate the elastic properties of collagenous tissues directly from noninvasive microscopy images. To our best knowledge, this is the first study in which Deep Learning techniques were used to derive nonlinear anisotropic elastic properties directly from tissue microscopy images. This work was motivated by the lengthy, complex, and destructive nature of traditional tissue mechanical testing. While it takes only about 10-30 minutes to obtain SHG images of a tissue sample, it takes much longer (hours) to prepare testing samples, set up testing and measurement instruments, and perform the actual mechanical test on each sample to obtain the stress-strain response curves. It took several months to obtain the data from the 48 samples used in this study. The success of this study holds

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

promise for the use of Machine Learning techniques to noninvasively and efficiently estimate the mechanical properties of many structure-based biological materials.

Traditional machine learning methods [17] require hand-engineered features (i.e. features defined by human experts), which are difficult to obtain for this application. Rezakhaniha *et al.* [44] have defined intuitive texture features of tissue fibers, such as waviness, straightness, bundle size, etc., but this requires time-consuming manual annotation. Moreover, it is unclear whether these hand-engineered features could fully describe the fiber network structural information. As an end-to-end solution, CNN eliminates the need for hand-engineered features. One factor limiting the use of CNN and Deep Learning methods in biomechanics applications, is that they generally require a large amount of training data [42, 43], while the sample size for mechanical testing of biological tissues is typically small, on the order of 10 – 100 samples. However, in this study, it is shown that the deep CNN can also work well with a small dataset by combining supervised and unsupervised learning methods, and utilizing data augmentation methods. As more images and mechanical testing data are collected, the performance of the CNN can be further improved.

The CNN architecture used in this study, was specifically designed for this application. The 1st to 5th block of the CNN serve as automatic feature extractors that convert the input image into a feature vector for classification and regression. The filters in the first convolution layer represent different local fiber network patterns, while the filters in the remaining convolution layers represent various combinations of these patterns at different locations and length scales. Two simplified versions of the CNN were tested, i.e., CNN-s1 and CNN-s2 with less filters and less layers. The results show that simplifications to the CNN led to a significant decrease of accuracy, which may be the result of signal loss during signal propagation due to the fewer filters, and disruption of the encoding mechanism due to the fewer layers, respectively.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

The CNN also demonstrated superiority over a simple image-feature based method to estimate the overall stiffness of collagen-based materials. Raub *et al.* [41] showed that the skewness of an image histogram was correlated to the collagen concentration and the Young's modulus of collagen gels. Therefore, a softmax classifier was built by using the skewness as the only input feature in this study. As demonstrated in the results (Figure 7), the CNN outperformed the softmax classifier by a large margin; and even the two simplified versions of the CNN performed better than the softmax classifier, which underscores the superiority of CNNs for automatically extracting fiber network features.

More importantly, we demonstrated that the CNN can predict the PCA parameters of the stress-strain curves, such that the entire anisotropic stress-strain response of GLBP tissues can be estimated. For a nonlinear elastic response, it is well known that the Young's modulus or stiffness cannot fully describe the tissue mechanical behavior, since the tangential value changes at different stress/strain levels along the nonlinear stress-strain curve. Thus, the PCA parameters offer a much more comprehensive look at the tissue elastic properties. Interestingly, we found that for this application, the overall "shape" of a stress-strain curve can be described with a single parameter, α_1 in Eq.(1). The novel PCA based approach to represent stress-strain curves developed in this study may facilitate more thorough analysis and comparison of tissue stress-strain responses over basic stiffness metrics.

This approach opens the door for the fast and noninvasive assessment of collagenous tissue elastic properties from microstructural images, enabling many potential applications such as serving as a quality control tool for the manufacturing of BHVs.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

5. CONCLUSION

In conclusion, this study demonstrated the feasibility of using Deep Learning techniques for fast and noninvasive assessment of collagenous tissue elastic properties from microcopy images. The main contributions of this study include: 1) the use of PCA to parameterize equi-biaxial stress-strain curves and quantify the overall stiffness, 2) the custom deep convolutional neural network design to automatically extract structural patterns of collagenous tissues, and perform classification to identify overall stiffness, as well as regression to predict PCA- parameters of nonlinear anisotropic stress-strain curves, and 3) the unsupervised deep learning method combined with supervised learning and data augmentation to overcome the challenge of small datasets for Deep Learning in the field of biomechanics. The developed approach was evaluated through cross validation, where an average classification accuracy of 84% and average regression errors of 0.021 and 0.031 were achieved. This study clearly demonstrates the great potential for Machine Learning techniques to estimate tissue mechanical properties solely through the use of noninvasive microcopy images.

ACKNOWLEDGEMENTS

Research for this project is funded in part by NIH grant R01 HL104080. Liang Liang is supported by an American Heart Association Post-doctoral fellowship 16POST30210003. The authors thank Fatiesa Sulejmani and Andres Caballero for assisting in the collection of biaxial testing data and SHG images used in this study, as well as Caitlin Martin for comments and suggestions.

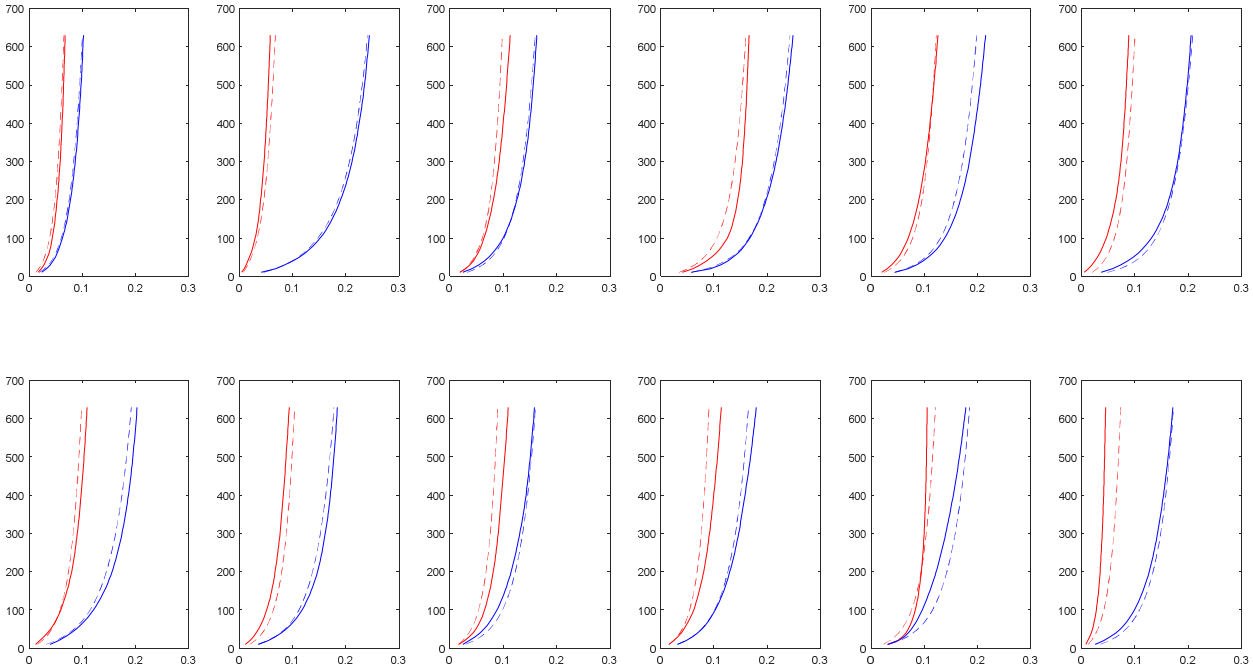
CONFLICT OF INTEREST STATEMENT

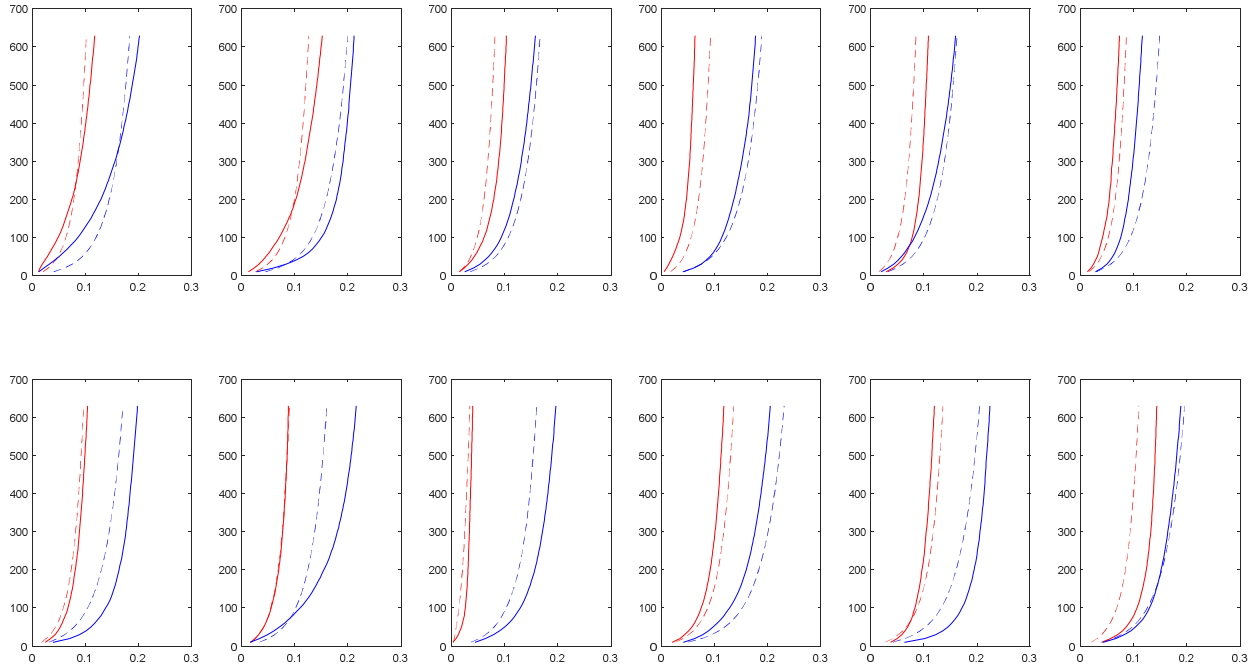
None.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

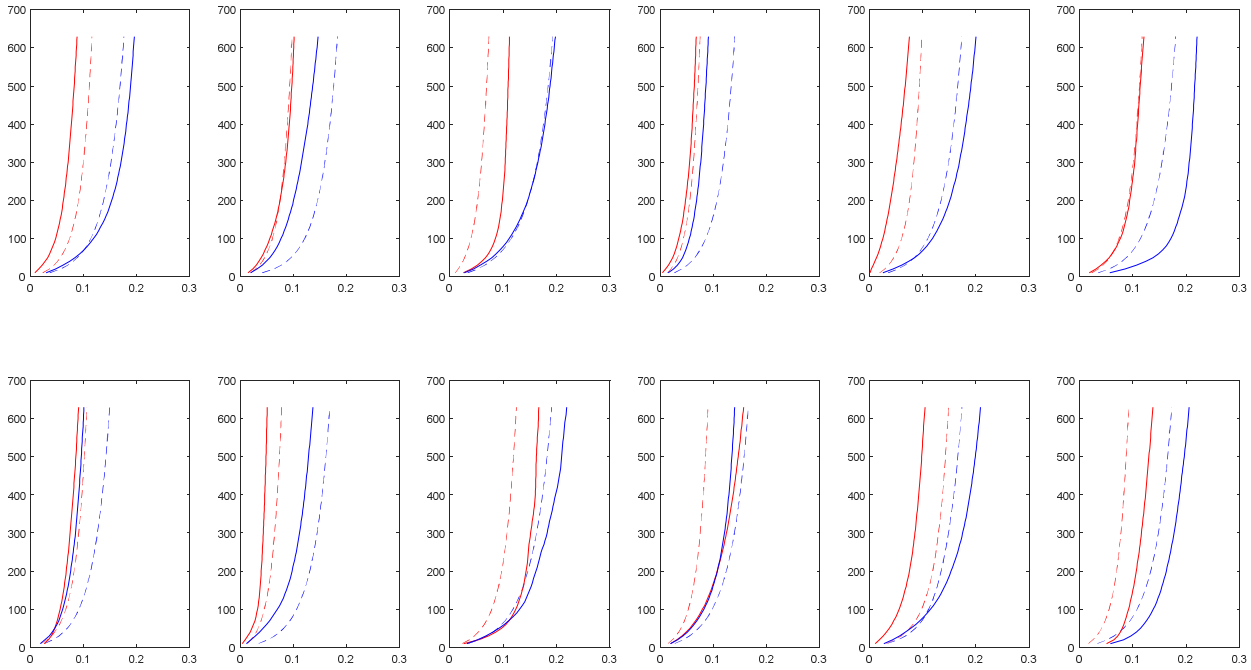
Appendix

Predicted stress-strain curves of the 48 tissue samples are shown from the best to the worst. Horizontal axis shows Green Strain. Vertical axis shows 2nd PK stress (*KPa*). Dashed lines are predicted stress-strain curves, and solid lines are the curves from mechanical testing.



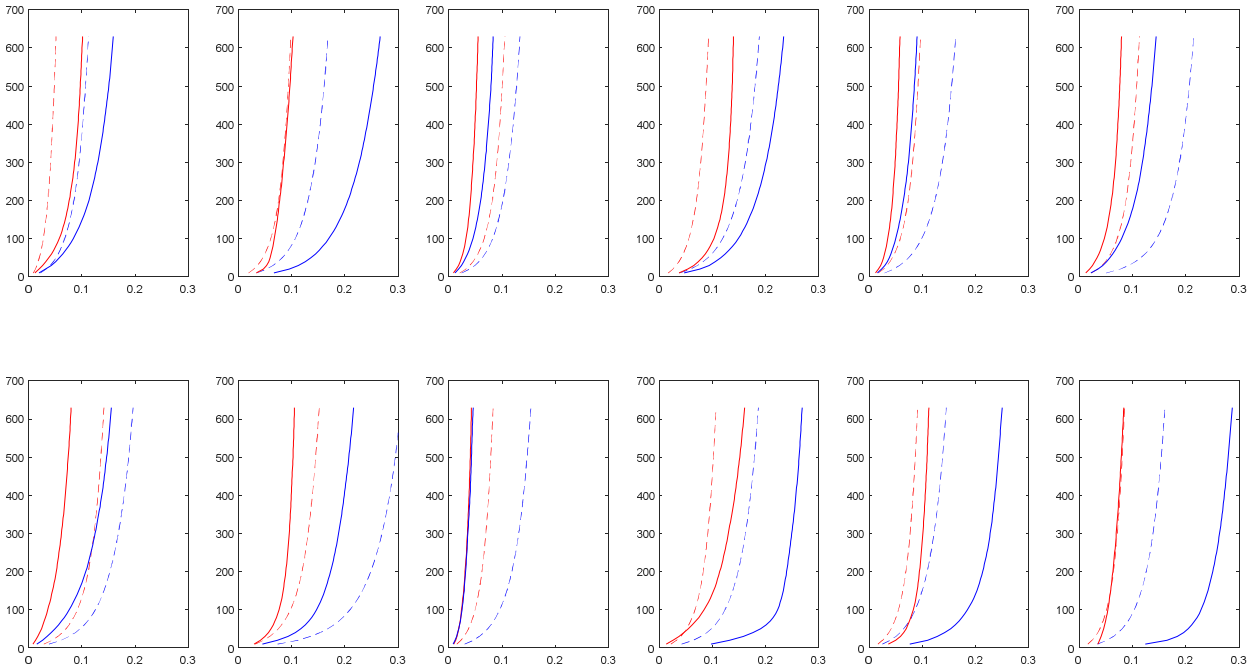


bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.



420

421



422

423

424

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

REFERENCES

[1] Fomovsky GM, Thomopoulos S, Holmes JW. Contribution of Extracellular Matrix to the Mechanical Properties of the Heart. Journal of molecular and cellular cardiology 2010;48:490-6.
[2] Sacks MS. Incorporation of Experimentally-Derived Fiber Orientation into a Structural Constitutive Model for Planar Collagenous Tissues. Journal of Biomechanical Engineering 2003;125:280-7.
[3] Nimni ME, Cheung D, Strates B, Kodama M, Sheikh K. Chemically modified collagen: A natural biomaterial for tissue replacement. Journal of Biomedical Materials Research 1987;21:741-71.
[4] Khor E. Methods for the treatment of collagenous tissues for bioprostheses. Biomaterials 1997;18:95-105.
[5] Vesely I. The evolution of bioprosthetic heart valve design and its impact on durability. Cardiovascular Pathology 2003;12:277-86.
[6] Lam MT, Wu JC. Biomaterial applications in cardiovascular tissue repair and regeneration. Expert review of cardiovascular therapy 2012;10:1039-49.
[7] Brown P. Abdominal Wall Reconstruction Using Biological Tissue Grafts. AORN Journal 2009;90:513-24.
[8] Demange MK, de Almeida AM, Rodeo SA. Updates in biological therapies for knee injuries: tendons. Current Reviews in Musculoskeletal Medicine 2014;7:239-46.
[9] Huerta S, Varshney A, Patel PM, Mayo HG, Livingston EH. Biological mesh implants for abdominal hernia repair: Us food and drug administration approval process and systematic review of its efficacy. JAMA Surgery 2016;151:374-81.
[10] Zhang L, Lake SP, Lai VK, Picu CR, Barocas VH, Shephard MS. A coupled fiber-matrix model demonstrates highly inhomogeneous microstructural interactions in soft tissues under tensile load. Journal of biomechanical engineering 2013;135:011008-.
[11] Jin T, Stanciulescu I. Computational modeling of the arterial wall based on layer-specific histological data. Biomech Model Mechanobiol 2016;15:1479-94.
[12] Jin T, Stanciulescu I. Numerical simulation of fibrous biomaterials with randomly distributed fiber network structure. Biomech Model Mechanobiol 2016;15:817-30.
[13] D'Amore A, Amoroso N, Gottardi R, Hobson C, Carruthers C, Watkins S, et al. From single fiber to macro-level mechanics: A structural finite-element model for elastomeric fibrous biomaterials. Journal of the Mechanical Behavior of Biomedical Materials 2014;39:146-61.
[14] Picu RC. Mechanics of random fiber networks-a review. Soft Matter 2011;7:6768-85.
[15] Liu Q, Lu Z, Hu Z, Li J. Finite element analysis on tensile behaviour of 3D random fibrous materials: Model description and meso-level approach. Materials Science and Engineering: A 2013;587:36-45.
[16] Wicker BK, Hutchens HP, Wu Q, Yeh AT, Humphrey JD. Normal basilar artery structure and biaxial mechanical behaviour. Comput Methods Biomech Biomed Engin 2008;11:539-51.
[17] LeCun Y, Bengio Y, Hinton GE. Deep Learning. Nature 2015;521:436-44.
[18] Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. Neural Information Processing Systems 2012.
[19] LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE 1998;86:2278-324.
[20] 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.
[21] Kokkinos I. Pushing the Boundaries of Boundary Detection using Deep Learning. Intl Conf on Learning Representations 2016.
[22] 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.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

[23] He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition 2016.
[24] Caballero A, Sulejmani F, Martin C, Pham T, Sun W. Evaluation of Transcatheter Heart Valve Biomaterials: Biomechanical Characterization of Bovine and Porcine Pericardium. Journal of Materials Science: Materials in Medicine (under review) 2017.
[25] Sacks MS, Sun W. Multiaxial Mechanical Behavior of Biological Materials. Annual Review of Biomedical Engineering 2003;5:251-84.
[26] Sun W, Sacks M, Fulchiero G, Lovekamp J, Vyavahare N, Scott M. Response of heterograft heart valve biomaterials to moderate cyclic loading. Journal of Biomedical Materials Research Part A 2004;69A:658-69.
[27] Sun W, Abad A, Sacks MS. Simulated Bioprosthetic Heart Valve Deformation under Quasi-Static Loading. Journal of Biomechanical Engineering 2005;127:905-14.
[28] Devijver PA. Pattern Recognition: A Statistical Approach. London, GB: Prentice-Hall; 1982.
[29] Heimann T, Meinzer H-P. Statistical shape models for 3D medical image segmentation: a review. Medical Image Analysis 2009;13:543-63.
[30] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. Proceedings of The 32nd International Conference on Machine Learning 2015:448-56.
[31] Glorot X, Bordes A, Bengio Y. Deep Sparse Rectifier Neural Networks. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics 2011.
[32] Vedaldi A, Lenc K. MatConvNet: Convolutional Neural Networks for MATLAB. Proceedings of the 23rd ACM international conference on Multimedia. Brisbane, Australia: ACM; 2015. p. 689-92.
[33] Jarrett K, Kavukcuoglu K, Ranzato MA, LeCun Y. What is the Best Multi-Stage Architecture for Object Recognition? International Conference on Computer Vision 2009.
[34] Lei Z, Yi D, Li SZ. Learning Stacked Image Descriptor for Face Recognition. IEEE Transactions on Circuits and Systems for Video Technology 2016;26:1685-96.
[35] Bengio Y, Lamblin P, Popovici D, Larochelle H. Greedy layer-wise training of deep networks. Proceedings of the 19th International Conference on Neural Information Processing Systems. Canada: MIT Press; 2006. p. 153-60.
[36] Hinton GE, Osindero S, Teh Y-W. A Fast Learning Algorithm for Deep Belief Nets. Neural Computation 2006;18:1527-54.
[37] Markovsky I. Structured low-rank approximation and its applications. Automatica 2008;44:891-909.
[38] Kooi T, Litjens G, van Ginneken B, Gubern-Mérida A, Sánchez CI, Mann R, et al. Large scale deep learning for computer aided detection of mammographic lesions. Medical Image Analysis 2017;35:303-12.
[39] Isensee F, Kickingereder P, Bonekamp D, Bendszus M, Wick W, Schlemmer H-P, et al. Brain Tumor Segmentation Using Large Receptive Field Deep Convolutional Neural Networks. In: Maier-Hein gFKH, Deserno gLTM, Handels H, Tolxdorff T, editors. Bildverarbeitung für die Medizin 2017: Algorithmen - Systeme - Anwendungen Proceedings des Workshops vom 12 bis 14 März 2017 in Heidelberg. Berlin, Heidelberg: Springer Berlin Heidelberg; 2017. p. 86-91.
[40] Liu S, Zheng H, Fengc Y, Lid W. Prostate Cancer Diagnosis using Deep Learning with 3D Multiparametric MRI. SPIE Medical Imaging International Society for Optics and Photonics.
[41] Raub CB, Putnam AJ, Tromberg BJ, George SC. Predicting bulk mechanical properties of cellularized collagen gels using multiphoton microscopy. Acta Biomaterialia 2010;6:4657-65.
[42] Deng J, Dong W, Socher R, Li LJ, Kai L, Li F-F. ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition2009. p. 248-55.
[43] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision 2015;115:211-52.

bioRxiv preprint doi: https://doi.org/10.1101/154161; this version posted June 23, 2017. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

518 [44] Rezakhaniha R, Agianniotis A, Schrauwen JTC, Griffa A, Sage D, Bouten CVC, et al. Experimental
519 investigation of collagen waviness and orientation in the arterial adventitia using confocal laser scanning
520 microscopy. Biomech Model Mechanobiol 2012;11:461-73.
521 [45] Cootes TF, Taylor CJ, Cooper DH, Graham J. Active shape models - their training and application.
522 Computer Vision and Image Understanding 1995;61:38-59.

523