Ultrasound Elastography using Machine Learning

Abdelrahman Zayed

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Electrical and Computer Engineering) at

Concordia University

Montréal, Québec, Canada

March 2020

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Abdelrahman Zayed**

Entitled: **Ultrasound Elastography using Machine Learning**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Electrical and Computer Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. O. Ait Mohamed

_____ External Examiner
Dr. A. Ben Hamza

_____ Examiner
Dr. O. Ait Mohamed

_____ Supervisor
Dr. H. Rivaz

Approved by _____
Yousef R. Shayan, Chair
Department of Electrical and Computer Engineering

_____ 2020

_____
Amir Asif, Dean
Faculty of Engineering and Computer Science

Abstract

Ultrasound Elastography using Machine Learning

Abdelrahman Zayed

This thesis aims at solving two main problems that we face in ultrasound elastography, namely fast strain estimation and radio frequency (RF) frame selection. We rely on machine learning concepts such as Principal Components Analysis (PCA), multi-layer perceptron (MLP) and convolutional neural networks (CNN) to build 3 models that are trained on both phantom and *in vivo* data. In our first work, we developed a method to estimate the initial displacement between two ultrasound RF frames using PCA. We first compute an initial displacement estimate of around 1% of the samples, and then decompose the displacement into a linear combination of principal components (obtained offline during the training step). Our method assumes that the initial displacement of the whole image could also be described by this linear combination of principal components. This yields the same result that we could have had if we run dynamic programming (DP). The advantage of using PCA is that we could compute the same initial displacement image more than 10 times faster than DP. We then pass the result to GLobal Ultrasound Elastography (GLUE) for fine-tuning it, so we call the method PCA-GLUE.

In our second work, we developed a novel method to address the problem of RF frame selection in ultrasound elastography. Intuitively, we would like to have a classifier that gives a binary 1 to RF frame pairs that yield high-quality strain images. We make use of our previous work where we decompose the initial displacement between two RF frames into a weight vector multiplied by some

principal components. We consider the weight vector as our input feature vector to an MLP model. Given two RF frames I_1 and I_2, the MLP model predicts the normalized cross correlation (NCC) between the two RF frames I_1 and $I_2{}'$ ($I_2{}'$ is I_2 after being displaced according to the displacement of GLUE/PCA-GLUE). Our final contribution in this line of research is the introduction of a CNN-based method for RF frame selection as follows. First, we changed the architecture from an MLP model to a CNN that takes the two RF frames on two channels. The CNN has better results compared to the MLP model due to having more features. Second, we improved the automatic labelling of the data by having physical conditions that must be satisfied together in order to consider the pair as a suitable pair of RF frames.

Acknowledgments

I would like to express my deep gratitude to my great supervisor, Professor Hassan Rivaz. Dr. Rivaz is a very caring, knowledgeable and inspiring professor. Thanks to his continuous support, I have had a great environment to work on and solve different engineering problems. I am also very grateful to my family for their love and all the sacrifice they made for me to succeed. I would like to thank my friends and my colleagues in IMPACT lab, who helped and supported me throughout my studies.

I am thankful to both Richard and Edith Strauss Foundation and Quebec Bio-imaging Network (QBIN) for funding this work. I would like to thank Drs. E. Boctor, M. Choti and G. Hager for providing us with the *in vivo* patient's data from Johns Hopkins Hospital. I also thank Morteza Mirzaei for his help in collecting the phantom data and for providing us with an optimized version of the normalized cross correlation (NCC) code, as well as the very fruitful discussions. Special thanks to Md Ashikuzzaman for his helpful comments and for providing us with the simulation data. Finally, I acknowledge NVIDIA for donating the graphics card.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Ultrasound Imaging

Ultrasound waves are sound waves with a frequency higher than 20 kHz, which is the upper audible limit of human hearing. Ultrasound imaging is an imaging modality that uses ultrasound waves lying in the range of 1 MHz – 20 MHz. Due to being real-time, simple and non-invasive, ultrasound imaging has become one of the most widely used imaging modalities with numerous applications such as detecting diseases and image-guided surgery.

Fig. 1.1 shows an ultrasound device at Concordia University's PERFORM Centre. One of the most important parts in the ultrasound device is the transducer probe, as it is responsible for transmitting and receiving ultrasound waves. For the ultrasound image to be formed, we place the probe on the desired organ and start applying an external force in the axial direction. Inside the probe, there is a piezoelectric crystal which converts the applied mechanical stress into electric charge (this is known as the piezoelectric effect). As a result, ultrasound waves are transmitted from the transducer towards the body. When the waves hit an object, some portion continues through the body and the rest is reflected to the transducer probe, and then converted to electrical current

(the piezoelectric effect is reversible). By knowing the speed of the ultrasound wave and the time elapsed between transmission and reception, we can compute the depth of the object that has been hit. Therefore, we can form an image of the desired region of the body as shown in Fig.1.2.

Figure 1.1: Alpinion E-CUBE 12R ultrasound system. Image courtesy of: https://site.ieee.org/ius-2016/sponsors-exhibits/alpinion/ecube12_p/.

The frequency of the ultrasound wave used depends on the desired penetration depth, where higher penetration is achieved by using low frequency waves and vice versa. The reason is that the higher the frequency of the wave is, the more attenuated it becomes. Depending on the depth and structure of the targeted region of the body, different types of probes and different frequencies are used. The types of transducer probes are linear, curvilinear and phased array probes, as shown in Fig. 1.3. Linear probes work at a relatively high frequency. They are therefore used for imaging the near surface of the body, for example vascular imaging and testicular assessment. Curvilinear probes work at a low frequency, which allows deep penetration. They are used for imaging the

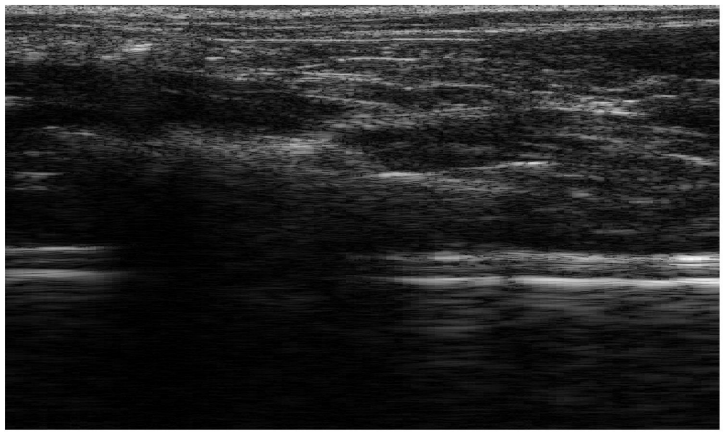


Figure 1.2: Ultrasound image collected by Md Ashikuzzaman at Concordia University's PERFORM Centre from a rat [1].

intra-abdominal structures as they have a wide depth of field. As for phased array probes, they also work at low frequency and are characterized by having a small footprint and a large depth of field, which allows them to image deep structures though a small acoustic window. They are ideal for imaging the chest by placing the probe between the ribs of the patient.

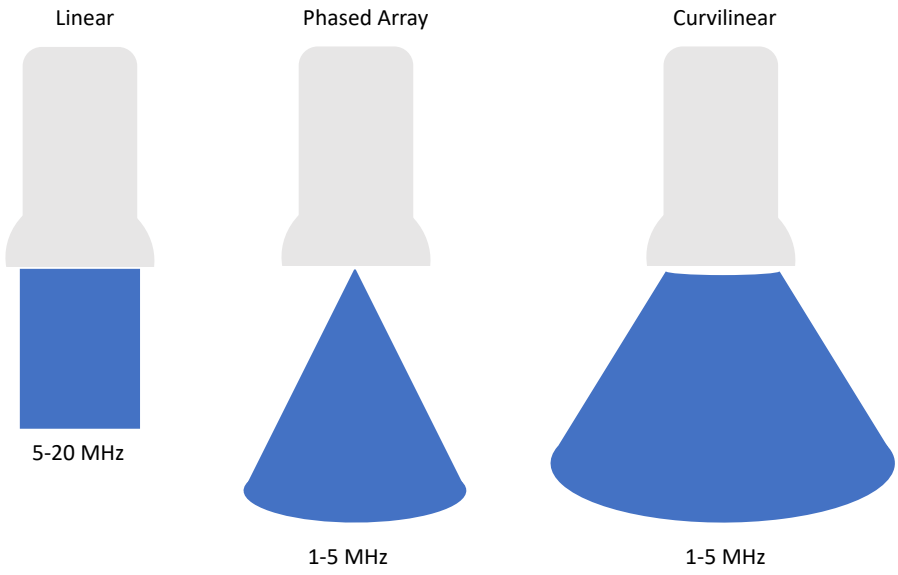


Figure 1.3: Different types of ultrasound probes and the range of frequencies used for each type.

1.1.1 Using deep learning in ultrasound imaging

After the success of deep learning in computer vision, it has been widely used in various medical imaging applications. In breast imaging, ultrasound plays a significant role in distinguishing malignant from benign lesions [2]. Even though this task could be done by different machine learning classifiers [3–5] without the use of deep learning, it has been shown that better results are obtained by using deep learning frameworks [6–8]. Computer-aided diagnosis (CADx) systems, which are used to provide an objective report to help radiologists with the interpretation and diagnosis of the medical image [9–18], are currently using deep learning to make their predictions. Due to the high risk associated with classifying a malignant tumor as benign, differentiating the cancerous from non-cancerous tumors is considered one of the most important applications of CADx systems [11, 12, 19–24].

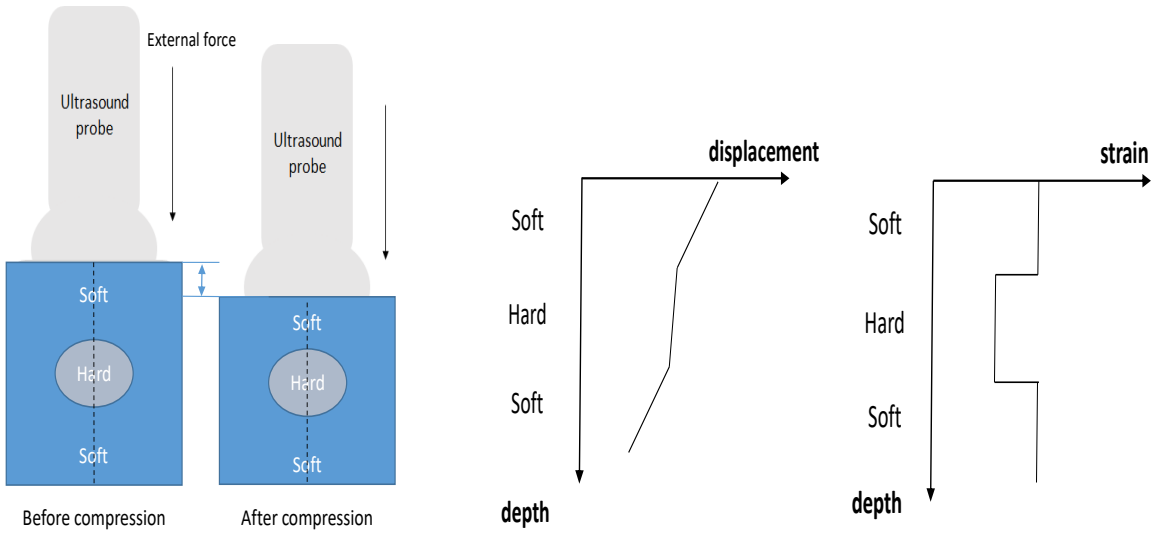
Deep learning can also be used for ultrasound image segmentation [25] to replace manual segmentation, thus saving both time and effort. Moreover, generative adversarial networks [26] have been used to increase the resolution of the ultrasound image without affecting the frame rate by having a multi-focus image [27].

1.2 Ultrasound Elastography

Ultrasound elastography is a branch of ultrasound that focuses on estimating the mechanical properties of the tissue such as strain. Elastography is used to distinguish cancerous from non-cancerous tissues. It is broadly classified into quasi-static elastography and dynamic elastography [28]. In this thesis, we focus on the quasi-static elastography.

1.2.1 Strain estimation

In order to estimate the strain image, we apply an external force on the desired organ, which causes some deformation in the tissues. During compression, the ultrasound device collects hundreds of radio frequency (RF) frames. We simply choose 2 RF frames that are collected before and after the deformation. We then estimate the displacement image, which determines the amount of motion that occurred in different parts of the tissue. By spatially differentiating the displacement image, we obtain the strain image as shown in Fig. 1.4.



(a) Data collection (b) Displacement (c) Strain

Figure 1.4: The basic steps of quasi-static ultrasound elastography. After data collection in (a), we estimate the amount of displacement in every sample in the RF frame, yielding the displacement image as shown in (b). Finally, we obtain the strain image in (c) by spatially differentiating the displacement image.

1.2.2 RF frame selection

One of the problems that face ultrasound elastography is choosing suitable RF frames that yield high-quality strain images. If the user who is responsible for applying the force is not experienced enough, the RF frames would have an out-of-plane deformation, yielding a low-quality strain image.

Therefore, we need to be very careful when choosing the pair of RF frames so that the motion is completely axial (i.e in-plane). To solve this problem, we introduced 2 different models in this thesis that can search for the best possible RF frame to be paired with a given pre-selected one, so as to have the best possible strain image.

1.3 Organization of the Thesis

We start in Chapter 1 of the thesis with an introduction to ultrasound imaging and specifically ultrasound elastography. In Chapter 2, we introduce PCA-GLUE for fast strain estimation and a multi-layer perceptron (MLP) classifier for RF frame selection. Chapter 3 introduces an improvement to the RF frame selection model by using a convolutional neural network (CNN). Chapter 4 concludes the whole thesis and discusses different ways to improve the results in the future. The appendices provide more simulation results in case the reader is interested in visualizing the effect of changing the hyperparameters of PCA-GLUE or testing the MLP classifier when all of the RF frames are out-of-plane.

1.4 Publications

The thesis has culminated in the following publications:

1. A. Zayed and H. Rivaz, Fast Approximate Time-Delay Estimation in Ultrasound Elastography Using Principal Component Analysis, IEEE 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2019, 6204-6207.

2. A. Zayed, H. Rivaz, Automatic Frame Selection Using MLP Neural Network in Ultrasound Elastography, ICIAR, Springer, Cham, 2019, pp 462-472.

3. A. Zayed and H. Rivaz, Fast Strain Estimation and Frame Selection in Ultrasound Elastography using Machine Learning, revision under review at IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control (TUFFC).

4. A. Zayed, G. Cloutier and H. Rivaz, Automatic Frame Selection using CNN in Ultrasound Elastography, IEEE 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2020, in press.

Chapter 2

Fast Strain Estimation and Frame Selection in Ultrasound Elastography using Machine Learning

As we discussed briefly in the previous chapter, ultrasound elastography aims to determine the mechanical properties of the tissue by monitoring tissue deformation due to internal or external forces. Tissue deformations are estimated from ultrasound radio frequency (RF) signals and are often referred to as time delay estimation (TDE). Given two RF frames I_1 and I_2, we can compute a displacement image which shows the change in the position of each sample in I_1 to a new position in I_2. Two important challenges in TDE include high computational complexity and the difficulty in choosing suitable RF frames. Selecting suitable frames is of high importance because many pairs of RF frames either do not have acceptable deformation for extracting informative strain images or are decorrelated and deformation cannot be reliably estimated. Herein, we introduce a method that learns 12 displacement modes in quasi-static elastography by performing Principal Component Analysis (PCA) on displacement fields of a large training database. In the inference

stage, we use dynamic programming (DP) to compute an initial displacement estimate of around 1% of the samples, and then decompose this sparse displacement into a linear combination of the 12 displacement modes. Our method assumes that the displacement of the whole image could also be described by this linear combination of principal components. We then use the GLobal Ultrasound Elastography (GLUE) method to fine-tune the result yielding the exact displacement image. Our method, which we call PCA-GLUE, is more than 10 times faster than DP in calculating the initial displacement map while giving the same result. This is due to converting the problem of estimating millions of variables in DP into a much simpler problem of only 12 unknown weights of the principal components. Our second contribution in this chapter is determining the suitability of the frame pair I_1 and I_2 for strain estimation, which we achieve by using the weight vector that we calculated for PCA-GLUE as an input to a multi-layer perceptron (MLP) classifier. We validate PCA-GLUE using simulation, phantom, and *in vivo* data. Our classifier takes only 1.5 *ms* during the testing phase and has an F1-measure of more than 92% when tested on 1,430 instances collected from both phantom and *in vivo* datasets.

2.1 Introduction

Ultrasound elastography has numerous applications in medical diagnosis of diseases and in image-guided interventions [29–36]. For example, it can be used in imaging cancer tumors by estimating the strain image since tumors are normally more rigid than the surrounding tissue. Ultrasound elastography has two main branches which are dynamic and quasi-static elastography [28]. Dynamic elastography refers to the quantitative estimation of the mechanical properties of the tissue. Quasi-static elastography, which is our focus in this thesis, is more related to estimating the deformation of the tissue when an external force is applied [37,38]. Recent work has shown success in performing ultrasound elastography using different methods such as spatial angular compounding

[39], multi-compression strategy [40], Lagrangian tracking [41] and guided circumferential waves [42]. In addition, other work has exploited the power of deep learning to achieve the same goal [43–50].

In spite of the various applications that ultrasound elastography has, it also has some challenges. One of these challenges is that TDE between frames of RF data is computationally expensive. The methods used for calculating the TDE are either optimization-based [51–53] or window-based [54–56]. In optimization-based techniques, the displacement image is estimated by minimizing a cost function. In window-based techniques, the objective is to find the displacement that maximizes a similarity metric such as normalized cross correlation (NCC) between two windows in the two frames before and after deformation.

Herein, we propose a computationally efficient technique for estimating an approximate TDE between two RF frames. To that end, we first learn the *modes* of TDE by acquiring a large training database of free-hand palpation elastography by intentionally compressing the tissue in different manners. We then perform PCA to extract the *modes* of TDE. At the test stage, we first run DP on only 1% of RF data to extract a sparse TDE between two frames I_1 and I_2. We then estimate the weights of principal components that best approximate this sparse TDE, and subsequently use the weighted principal components as an initial TDE for GLUE [51]. We therefore call our method PCA-GLUE. PCA-GLUE was inspired by the success of [57] in natural images. Similar work by Pohlman and Varghese [58] has shown promising results on displacement estimation using dictionary representations.

Another challenge that ultrasound elastography faces is the suitability of the RF frames to be used for strain estimation. The two RF frames used are collected before and after applying an external force. Depending on the direction of the applied force, different qualities of strain images would be obtained. To be more precise, in-plane displacement results in high-quality strain images,

whereas out-of-plane displacement results in low-quality strain images [59, 60]. This means that collecting ultrasound data needs the person to be experienced in applying purely axial force. For imaging some organs, it is hard to hold the probe and apply a purely axial force even for experts. Furthermore, even for pure axial compression, two RF frames can be decorrelated due to internal physiological motions, rendering accurate TDE challenging.

Many solutions have been introduced for solving this problem. Lubinski *et al.* [61] suggested averaging several displacement images to improve the quality. The weights used are not equal, they rather depend on the step size (i.e. certain images would have higher weights than others). Hiltawsky *et al.* [62] tried to tackle the out-of-plane displacement by developing a mechanical compression applicator to force the motion to be in-plane. Jiang *et al.* [63] defined a metric that informs the user whether or not to trust the pair of RF frames for strain estimation. This metric is the multiplication of the NCC of the motion compensated RF field and the NCC of the motion compensated strain field. Other approaches [64, 65] used an external tracker so as to pick up the RF frames that are collected roughly from the same plane. They used the tracking data to find pairs that have the lowest cost according to a predefined cost function. Ranger et al. [66] used a 3D camera to track and compensate any undesired motion that could happen during the data collection. Aalamifar et al. [67] used a robot for collecting RF frames. They try to estimate a transformation matrix that transforms the RF frames collected from the robot's tooltip to the ultrasound image frame, using an active echo element.

Although all of the previously mentioned approaches showed an improvement in the quality of the strain image, they also have some drawbacks. The approaches introduced in [62, 64–67] need an external device such as a mechanical applicator, a robot, a 3D camera or an external tracker. This not only complicates the process of strain estimation, but also makes it more expensive. The approach introduced in [63] gives a feedback on the quality of the strain image only *after estimating TDE*,

which means that it is not a computationally efficient method for frame selection. The method we propose in this chapter selects suitable frames *before estimating TDE* and is also computationally efficient.

Herein, we introduce a new method with three main contributions, which can be summarized as follows:

1. We develop a fast technique to compute the initial displacement image between two RF frames, which is the step prior to the estimation of the exact displacement image. Our method could also be used to speed up different displacement estimation methods by providing initial estimates.

2. We introduce a classifier that gives a binary decision for whether the pair of RF frames is suitable for strain estimation in only 1.5 *ms* on a desktop CPU.

3. PCA-GLUE, which relies on DP to compute the initial displacement map, is robust to potential DP failures.

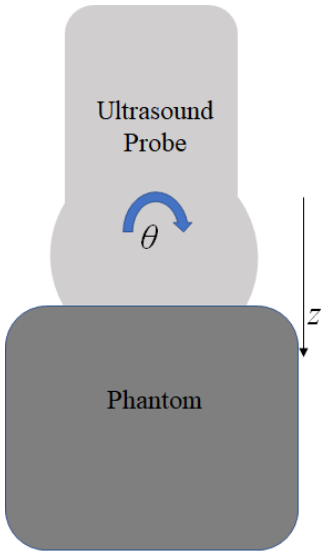
This is an extension of our recent work [68, 69], with the following major changes. First, we replace the MLP classifier with a more robust one that can generalize better to unseen data. Second, we used automatically annotated images for training the classifier, compared to manual annotation that we previously used in [69]. Third, testing is now substantially more rigorous and is performed on 5 different datasets from simulation, phantom and *in vivo* data. And last, the criteria for measuring the performance of the classifier used in this work are the accuracy and F1-measure instead of using the signal to Noise Ratio (SNR) and Contrast to Noise Ratio (CNR) in [69].

2.2 Methods

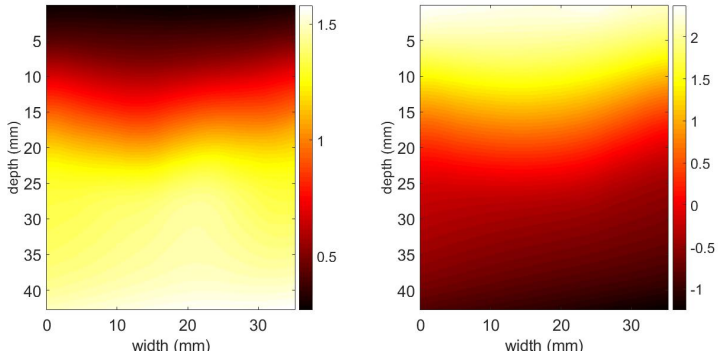
In this work, we have two main objectives which are fast TDE and automatic frame selection. We first propose a method that computes a superior approximate TDE compared to DP [70], while being more than 10 times faster.

The idea is simple and logical: we compute N principal components denoted by $\mathbf{b}_1$ to $\mathbf{b}_N$ from real experiments that describe TDE under the effect of an external force. In other words, the approximate displacement image is a linear combination of these principal components. During data collection, we applied the force in the 6 degrees of freedom (DOF) to ensure generality and a dataset of displacement images was obtained using GLUE. Using PCA, we were able to compute our principal components. Fig. 2.1 shows the directions of the applied force as well as some of the principal components learned.

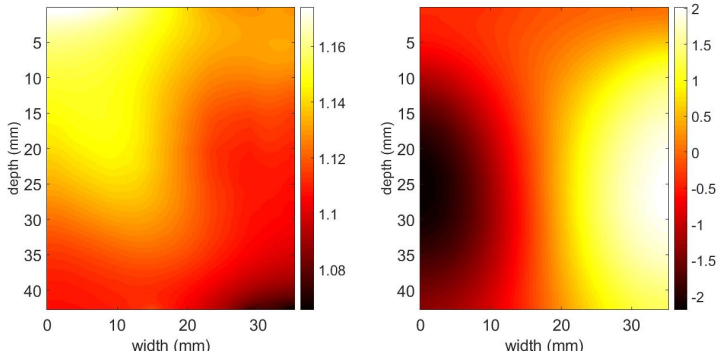
For frame selection, our goal was simply to have a classifier that can classify whether the two RF frames are suitable for strain estimation. One can consider an approach of having a classifier that takes the two RF frames, where the samples are the input features (such as [45] and [47]), and outputs a binary decision of 1 for suitable frames and 0 otherwise. This approach would need a powerful GPU as the number of samples in each RF frame is approximately 1 million. To simplify the problem, we make use of our representation for the displacement image by the principal components. We can think of this as a dimensionality reduction method for the huge number of features that we had, where the input feature vector can be simply the N-dimensional weight vector $\mathbf{w}$, which represents the weight of each principal component in the initial displacement image. Our low-dimensional weight vector $\mathbf{w}$ is the input to a multi-layer perceptron (MLP) classifier that would output a binary number 1 or 0 depending on whether the two RF frames are suitable or not for strain estimation.



(a) Directions of applied force

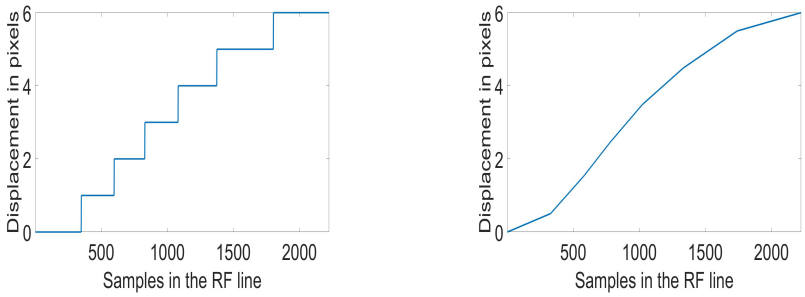


(b) Axial deformation (z)



(c) In-plane rotation (θ)

Figure 2.1: Principal components of in-plane axial displacement (in *mm*) learned from both *in vivo* and phantom experiments. In (a), translation of the probe along z and its rotation by θ generates axial deformation in the phantom. In (b), extension and compression principal components along z are shown. In (c), displacement arising from rotation by θ is shown.



(a) Before interpolation (b) After interpolation

Figure 2.2: The displacement of a certain RF line before and after interpolation.

2.2.1 Feature extraction

Consider having two RF frames I_1 and I_2 collected before and after some deformation, each of size $m \times l$, where m is the number of samples in an RF-line and l is the number a RF lines. Our goal is to estimate a coarse displacement image that describes the axial motion that each sample has had [71]. We start by running the DP algorithm on only p RF lines out of the total l RF lines (where $p << l$) to get the integer displacement of $k = m \times p$ pixels. We then form a k-dimensional vector named $\mathbf{c}$ after applying a simple linear interpolation to the k estimates to make them smoother, so that the integer estimates become linearly increasing with depth instead of the staircase approximation, as shown in Fig. 2.2.

Next, we construct the matrix $\mathbf{A}$ such that

$$\mathbf{A} = \begin{bmatrix} \mathbf{b}_1(q_1) & \mathbf{b}_2(q_1) & \mathbf{b}_3(q_1) & \dots & \mathbf{b}_N(q_1) \\ \mathbf{b}_1(q_2) & \mathbf{b}_2(q_2) & \mathbf{b}_3(q_2) & \dots & \mathbf{b}_N(q_2) \\ \dots & \dots & \dots & \dots & \dots \\ \mathbf{b}_1(q_K) & \mathbf{b}_2(q_K) & \mathbf{b}_3(q_K) & \dots & \mathbf{b}_N(q_K) \end{bmatrix} \tag{1}$$

where the N vectors from $\mathbf{b}_1$ to $\mathbf{b}_N$ represent our N principal components, q_1 to q_K correspond to our 2D coordinates of the sparse features chosen along the p RF lines before deformation. For example,

for an RF frame of size 2304×384, if we set p to 1 and choose the sparse features to be along the RF line number 200. Then $k = 2304$ and q_1 to q_K would be $\{(1,200),(2,200),\ldots..,(2304,200)\}$.

Next, we compute the weight vector $\mathbf{w} = (w_1,...,w_N)^T$ according to the following equation

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} ||\mathbf{Aw–c}|| \tag{2}$$

This implies that we choose the weight vector $\mathbf{w}$ that decomposes the actual displacement image into a linear combination of the principal components weighted by some coefficients so as to have the minimum sum-of-squared error.

2.2.2 Implementation

2.2.2.1 Implementing PCA-GLUE for strain estimation

Strain estimation relies on the extracted features to calculate the integer displacement image $\hat{\mathbf{d}}$.

$$\hat{\mathbf{d}} = \sum_{n=1}^{N} \hat{w_n}\mathbf{b}_n \tag{3}$$

Eq. 3 shows how to calculate the integer displacement image $\hat{\mathbf{d}}$ from the weight vector $\mathbf{w}$, which is then passed to GLUE to obtain the exact displacement image $\mathbf{d}$. Finally, the resulting image is spatially differentiated to obtain the strain image. Algorithm 1 summarizes the procedure followed by PCA-GLUE.

2.2.2.2 Implementing the MLP classifier for frame selection

The MLP classifier takes the weight vector $\mathbf{w}$ (see Algorithm 1 steps 2 to 4) as the input feature vector. The ground truth (i.e. whether I_1 and I_2 are suitable for strain estimation or not) is obtained according to the procedure described in Algorithm 2.

Algorithm 1 PCA-GLUE

procedure PCA-GLUE
    Choose p equidistant RF lines.
    Run DP to get the integer axial displacement of the p RF lines.
    Solve Eq. 2 to get the vector $\mathbf{w}$.
    Compute the initial axial displacement $\hat{\mathbf{d}}$ of all RF lines by Eq. 3.
    Use GLUE to calculate the exact axial displacement.
    Strain is obtained by spatial differentiation of the displacement.
end procedure

Algorithm 2 Labelling the dataset for the MLP classifier

procedure
    RF frames I_1 and I_2 are passed to GLUE to obtain the displacement image.
    I_2 is deformed and interpolated according to the computed displacement image yielding $I_2{}'$.
    We calculate the Normalized cross correlation (NCC) between I_1 and $I_2{}'$.
    The final decision is 1 if the NCC is higher than 0.9 and 0 otherwise.
end procedure

The issue with this algorithm is that it is slow because of three computationally expensive steps of 2, 3 and 4. As such, it cannot be performed on many pairs of RF frames in real-time. Our goal is to train a classifier that predicts the output of step 5 by bypassing steps 2 to 4. The architecture of our classifier is relatively simple, with an input layer, 3 hidden layers, and an output layer. The input layer takes the N-dimensional vector $\mathbf{w}$. The 3 hidden layers contain 256, 128 and 64 hidden units with a Rectified Linear Unit (ReLU) as the activation function. The output layer contains one unit, where the predicted value corresponds to the Normalized cross correlation (NCC) between I_1 and $I_2{}'$ such that

$$NCC = \frac{\sum_i (I_1(i) - \bar{I_1})(I_2(i)' - \bar{I_2}')}{\sqrt{\sum_i (I_1(i) - \bar{I_1})^2 \sum_i (I_2(i)' - \bar{I_2}')^2}} \forall i \in I_1 \cap I_2{}' \tag{4}$$

where $\bar{I_1}$ and $\bar{I_2}'$ are the mean values of the RF frames I_1 and $I_2{}'$ respectively.

NCC has been widely used as a similarity metric by several image registration methods [72–75]. In this work, we claim that the NCC between I_1 and $I_2{}'$ is an indicator for the suitability of I_1 and I_2

for elastography. Therefore, we apply a threshold on the value of both the predicted NCC and the ground truth NCC to compute the binary equivalent, which is 1 when the NCC is higher than 0.9 and 0 otherwise. One possible criticism to our work might be that we do not directly estimate the binary output. This is because better results were obtained when training is done to estimate the NCC, as opposed to training to obtain a binary decision. The reason is that the NCC value provides more information to the network for better training compared to its thresholded binary number. It also makes the derivative of the loss function smoother, resulting in improved backpropagation. Another benefit is to be able to pick up the best possible frame to be paired with a certain frame, where we pair it with the frame with the highest NCC in a specified window of the 16 nearest frames, which has only one solution (assuming that there exist good frames in the window), compared to multiple solutions if the result is just a binary number. Our loss function is the mean square error (MSE) between the estimated NCC and the actual NCC before thresholding. We use Adam optimizer [76] with a learning rate of $1e{-}3$. The code is written in Python using Keras [77]. Fig. 2.3 shows the overall procedure followed by our algorithm for frame selection. Fig. 2.4 contains a flowchart that shows how strain estimation and frame selection are augmented together.

2.2.3 Data Collection

2.2.3.1 PCA-GLUE

We collected 4,055 RF frames from 3 different CIRS phantoms (Norfolk, VA), namely Models 040GSE, 039 and 059 at different locations at Concordia University's PERFORM Centre. Model 040GSE has 3 different cylindrical regions with elasticity moduli of 10, 40 and 60 kPa. The 039 and 059 models have spherical inclusions that are distributed throughout the phantoms. The elasticity moduli of the inclusions are 27 kPa for Model 039 and in the range of 10-15 kPa for model 059. The compression was applied in 3 different directions: in-plane axial motion, in-plane rotation and

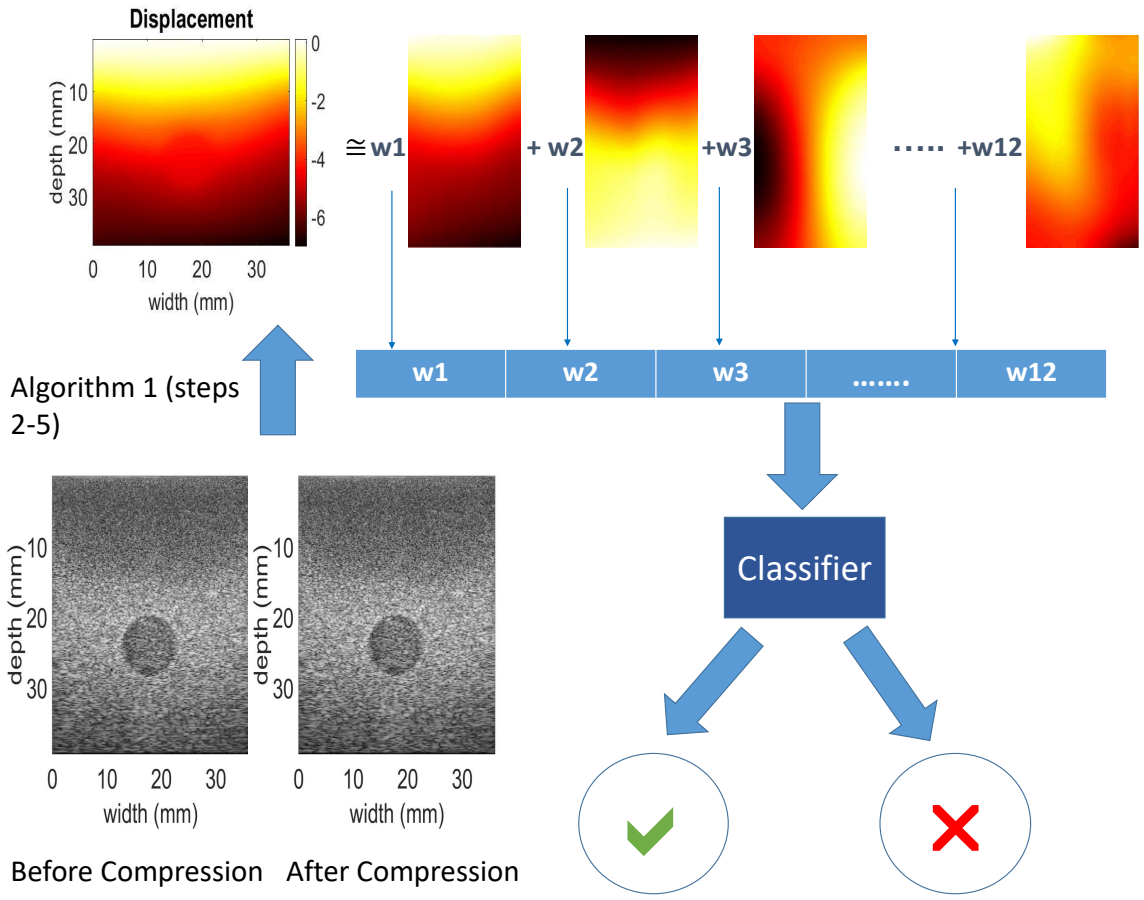


Figure 2.3: The overall procedure used for frame selection. Given two RF frames (we are showing here the B-mode images for illustration) collected before and after deformation, we first estimate the integer displacement image $\hat{\mathbf{d}}$ (in *mm*) using PCA-GLUE, by applying Algorithm 1 (steps 2-5). We then use the weight vector $\mathbf{w}$ as the input feature vector to the MLP classifier.

out-of-plane lateral motion. The ultrasound device used is the 12R Alpinion Ultrasound machine (Bothell, WA) with an L3-12H high density linear array probe at a center frequency of 8.5 MHz and sampling frequency of 40 MHz.

We also have access to 298 RF frames collected at Johns Hopkins Hospital from 3 different patients who were undergoing liver ablation for primary or secondary liver cancers using Antares Siemens system (Issaquah, WA) at a center frequency of 6.67 MHz with a VF10-5 linear array at a sampling rate of 40 MHz. The study has the approval of the institutional review board and an informed consent was obtained from the patients. 3,635 RF frames out of the total 4,055 phantom

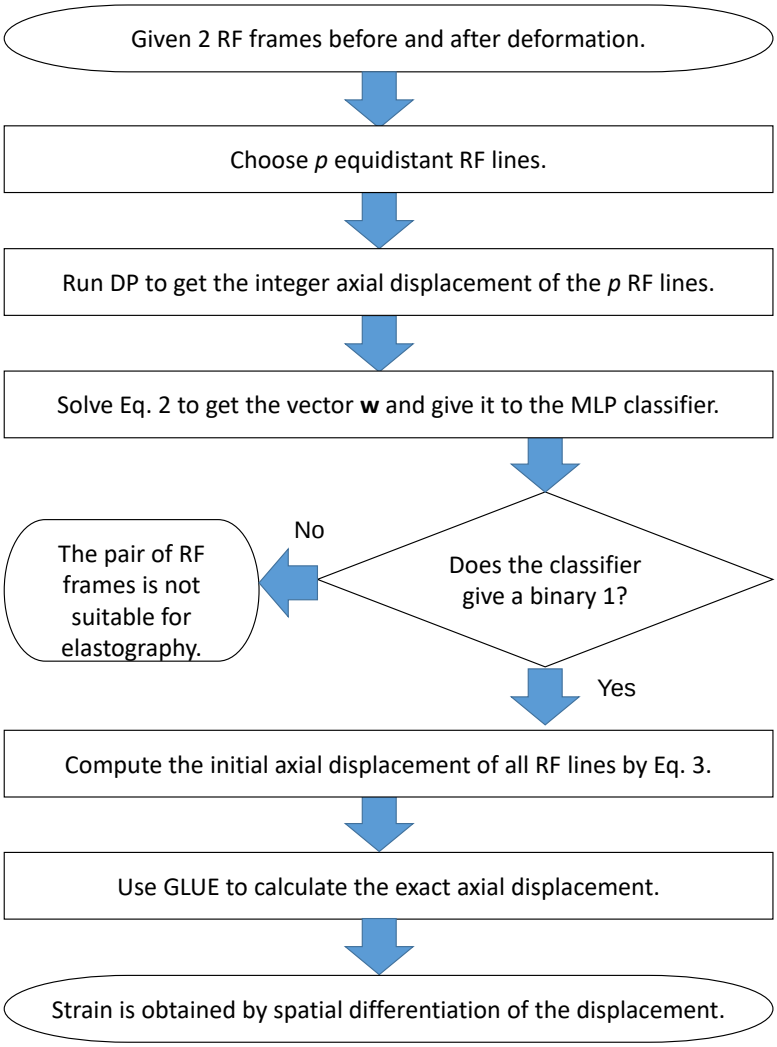


Figure 2.4: Flowchart of RF frame selection and strain estimation.

RF frames, along with 137 *in vivo* RF frames out of the total 298 *in vivo* RF frames were used for obtaining the principal components $\mathbf{b}_1$ to $\mathbf{b}_N$ by following the procedure in Algorithm 3, leaving 420 phantom RF frames and 161 *in vivo* RF frames for validating our method. It is important to note that the training data was excluded from further evaluation.

The simulation data was generated using Field II software [78, 79]. ABAQUS (Providence, RI) software was used to apply some compression, and the ground truth displacement was generated using finite element method (FEM).

2.2.3.2 MLP classifier

We used the data that we collected for PCA-GLUE for training our MLP classifier. It was trained on 4,662 instances from both phantom and *in vivo* data, which were partitioned between training and validation with a ratio 80:20. Testing was done on a different dataset composed of 1,430 frames. The ground truth was obtained by following the procedure in Algorithm 2.

2.2.4 Metrics used for performance assessment

In order to be able to quantitatively measure the performance of the strain estimation algorithm PCA-GLUE, we use two quality metrics which are the SNR and CNR [80], such that:

$$CNR = \frac{C}{N} = \sqrt{\frac{2(\bar{s_b} - \bar{s_t})^2}{\sigma_b^2 + \sigma_t^2}}, SNR = \frac{\bar{s}}{\sigma} \tag{5}$$

where $\bar{s_t}$ and σ_t^2 are the strain average and variance of the target window, $\bar{s_b}$ and σ_b^2 are the strain average and variance of the background window respectively. We use the background window for SNR calculation (i.e. $\bar{s}$=$\bar{s_b}$ and σ=σ_b). The background window is chosen in a uniform region where the strain values do not vary considerably. It is worth mentioning that the SNR and CNR values are obtained as the average over 10 different experiments.

Precision and recall are two important metrics for assessing the performance of a classifier. The F1-measure incorporates both metrics as follows:

$$F1 - measure = 2\frac{(Precision \times Recall)}{(Precision + Recall} \tag{6}$$

Algorithm 3 Obtaining the principal components

1: **procedure**
2: Run GLUE on the 3,772 RF frame pairs collected (3,635 from the phantom dataset and 137 from the *in vivo* dataset), yielding 3,772 displacement images.
3: Reshape every displacement image from a 2304×384 matrix into an $884,736 \times 1$ vector.
4: Form the data matrix $\mathbf{X}$ of size $884,736 \times 3,772$ by concatenating the 3,772 vectors.
5: Compute the covariance matrix as follows: $\mathbf{S} = \frac{1}{n} \times \mathbf{X}' \times \mathbf{X}'^T$, where $\mathbf{X}'$ is the matrix $\mathbf{X}$ after subtracting the mean value of the elements in each row (we set n to 3,772).
6: Obtain the eigenvalues of the matrix $\mathbf{S}$ and sort them descendingly.
7: Compute the eigenvectors corresponding to the largest 12 eigenvalues.
8: Obtain the 12 principal components for the axial displacement (Fig. 1 (b) and (c)) by reshaping each of the 12 eigenvectors from an $884,736 \times 1$ vector into a $2,304 \times 384$ matrix.
9: **end procedure**

2.3 Results

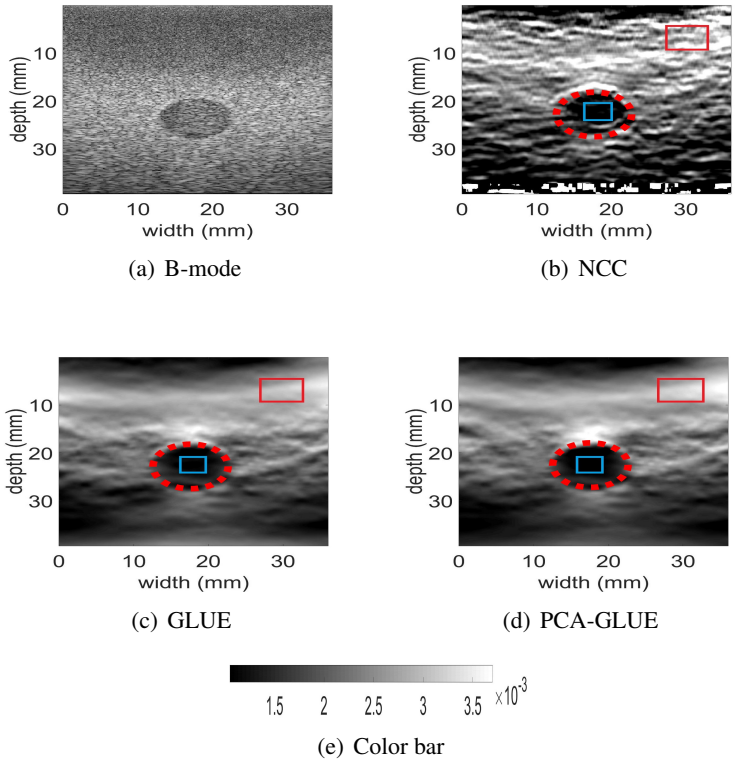
For our results, we set $N = 12$. This means that every displacement image is represented by 12 axial principal components in the form of a 12-dimentional vector $\mathbf{w}$. For results with different number of principal components, please refer to Appendix A of this thesis. We found that this representation captures 95% of the variance in the original data. For the NCC method, we used windows of size (5.42×12.49) λ. For DP estimation, the tunable parameter α_{DP} is set to 0.2. For GLUE, the parameters used during phantom experiments are $\alpha_1 = 5$, $\alpha_2 = 1$, $\beta_1 = 5$ and $\beta_2 = 1$. During *in vivo* experiments, we change GLUE's parameters to $\alpha_1 = 20$, $\alpha_2 = 1$, $\beta_1 = 20$ and $\beta_2 = 1$, to account for the increased noise. For NCC, GLUE and PCA-GLUE, the strain image is obtained from the displacement image using least square strain estimation [81].

For the running time, we trained PCA-GLUE in 5 hours, but training is done only once. For testing, we estimate the initial displacement in just 258 *ms* for two very large RF frames of sizes 2304×384 using an 8th generation 3.2 GHz Intel core i7 compared to 2.6 seconds if we use DP. For the frame selection, feature extraction and labeling the data took 30 hours, which included the procedure in Algorithm 2. The actual training of the MLP classifier took 29.16 seconds, while testing takes only 1.5 *ms*.

2.3.1 Phantom Results

2.3.1.1 Strain Estimation

Fig. 2.5 shows a comparison between the strain estimated using NCC, GLUE and PCA-GLUE for the phantom experiment, where the dashed circles point to the inclusion. The results of GLUE and PCA-GLUE look almost the same, but the advantage of using PCA-GLUE is that it estimates the initial estimates more than 10 times faster. Table 2.1 shows the SNR and CNR values obtained using different methods.



(a) B-mode (b) NCC

(c) GLUE (d) PCA-GLUE

(e) Color bar

Figure 2.5: The B-mode ultrasound and axial strain image using NCC, GLUE and PCA-GLUE for the real phantom experiment. The target and background windows are used for calculating SNR and CNR. The color bar is for the strain images only.

Table 2.1: The SNR and CNR values of the axial strain images for the phantom experiment. Target windows and background windows are of size 3 *mm* × 3 *mm* and 5 *mm* × 5 *mm* respectively as shown in Fig. 2.5. SNR is calculated for the background window.

Method used	**SNR**	**CNR**
NCC	18.18	16.86
GLUE	22.31	20.65
PCA-GLUE	**23.52**	**21.46**

2.3.1.2 Frame Selection

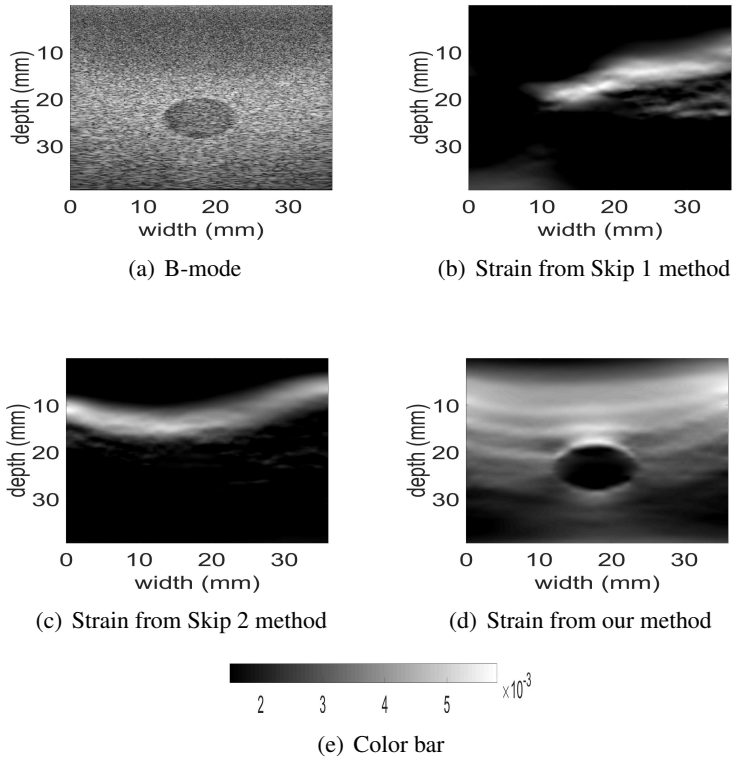
Our frame selection algorithm is compared to the simple method that chooses the pair of RF frames such that they are one or two frames apart. Fig. 2.6 shows the difference between applying our method and the fixed skip frame pairing while using PCA-GLUE for strain estimation. Our method considers a window of 16 frames, 8 of them are before the desired frame and 8 are after it. To choose a good frame to be paired with the desired frame, we run the MLP model on the 16 pairs and choose the pair that has the highest NCC (we don't apply the thresholding here). We can observe that our method selects RF frames that are suitable for strain estimation and it substantially outperforms the fixed skip frame pairing methods such as Skip 1 and Skip 2.

To make the validation more concrete, we test our classifier on 353 instances to classify them as suitable or not suitable for strain estimation. The ground truth is obtained as previously discussed in Algorithm 4. Table 2.2 shows the accuracy and F1-measure for our classifier on new data that the model has not seen before. The results show that our classifier is able to generalize well to unseen data, and that it could be used in practice.

2.3.2 *In vivo* Results

2.3.2.1 Strain Estimation

Fig. 2.7 and 2.8 show the results obtained when running NCC, GLUE and PCA-GLUE on the liver dataset, where both GLUE and PCA-GLUE yield very similar results. The dashed ellipses point to



(a) B-mode

(b) Strain from Skip 1 method

(c) Strain from Skip 2 method

(d) Strain from our method

(e) Color bar

Figure 2.6: The B-mode ultrasound and PCA-GLUE axial strain image for the phantom experiment using different frame selection methods. Note that the pair of RF data used for estimating strain is different from that of Fig. 2.5. The color bar is for the strain images only.

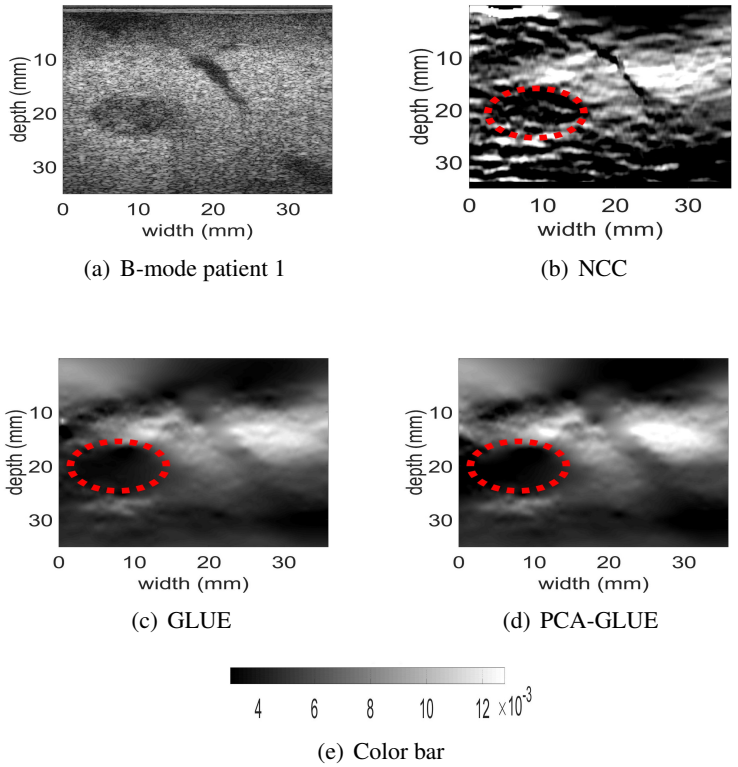
Table 2.2: The accuracy and F1-measure of our classifier on the phantom and *in vivo* test data.

Dataset	**Size**	**Accuracy**	**F1-measure**
Phantom	353 instances	85.11%	93.20%
Patient 1	147 instances	89.74%	96.86%
Patient 2	707 instances	70.43%	93.2%
Patient 3	223 instances	91.58%	92.52%

the tumors. Tables 2.3 and 2.4 show the SNR and CNR calculated.

Table 2.3: The SNR values of the axial strain images for the *in vivo* data.

Dataset	**NCC**	**GLUE**	**PCA-GLUE**
Patient 1	13.23	21.11	**21.19**
Patient 2	2.09	**21.33**	21.20
Patient 3	13.21	**25.66**	23.94



(a) B-mode patient 1

(b) NCC

(c) GLUE

(d) PCA-GLUE

(e) Color bar

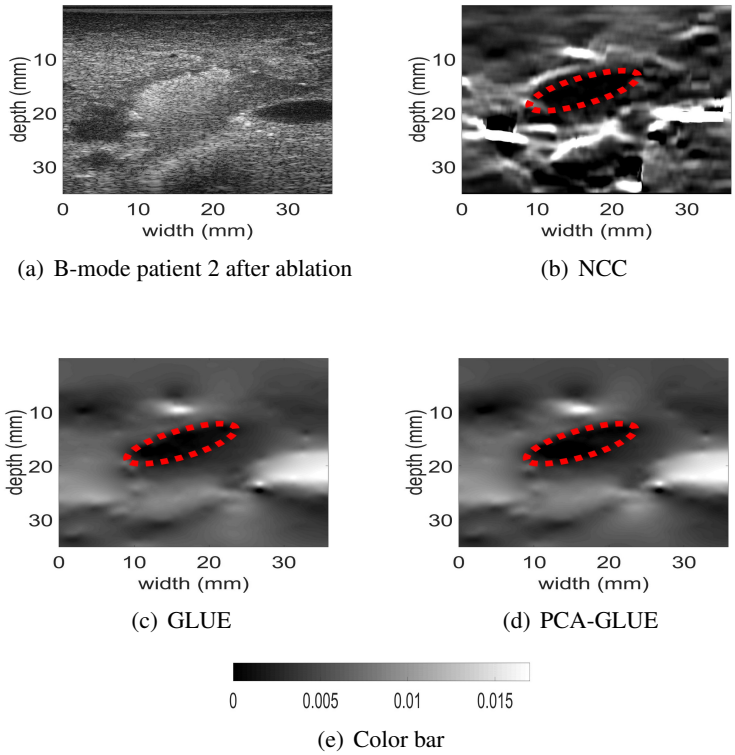
Figure 2.7: The B-mode ultrasound and axial strain image using NCC, GLUE and PCA-GLUE for the *in vivo* liver data before ablation. The color bar is for the strain images only.

Table 2.4: The CNR values of the axial strain images for the *in vivo* liver data.

Dataset	**NCC**	**GLUE**	**PCA-GLUE**
Patient 1	11.01	20.34	**20.66**
Patient 2	-0.46	13.52	**17.05**
Patient 3	9.87	**16.66**	15.95

2.3.2.2 Frame Selection

Fig. 2.9 shows a comparison between the strain estimated using both our frame selection method and the fixed skip frame pairing on two RF frames collected from the *in vivo* liver data. Table 2.2 shows the accuracy and F1-measure obtained for the liver dataset.



(a) B-mode patient 2 after ablation (b) NCC

(c) GLUE (d) PCA-GLUE

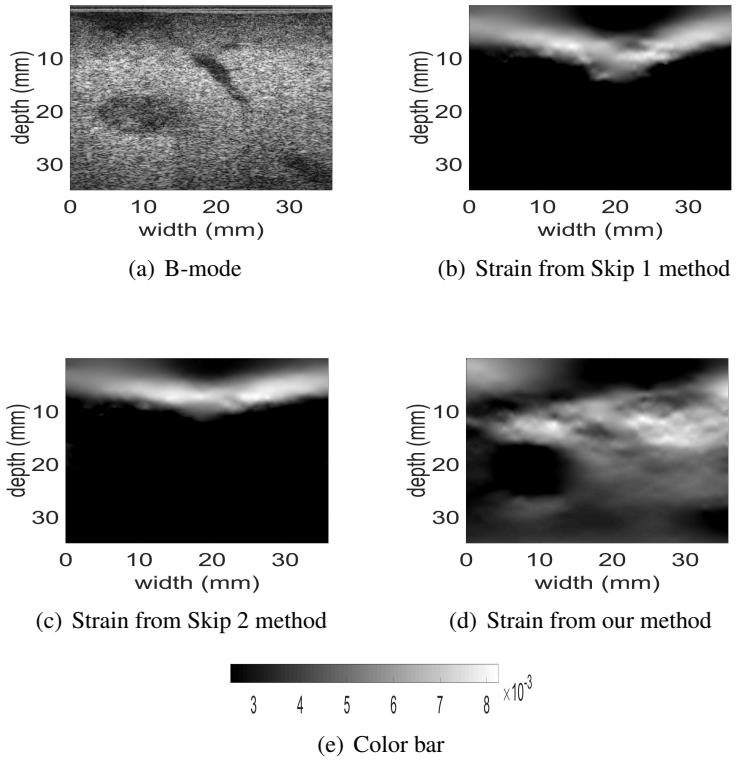
(e) Color bar

Figure 2.8: The B-mode ultrasound and axial strain image using NCC, GLUE and PCA-GLUE for the *in vivo* liver data after ablation. The color bar is for the strain images only.

2.3.3 PCA-GLUE robustness

Our method is not only capable of estimating strain or selecting suitable RF frames, it is also robust to incorrect initial displacement estimates when DP fails. The main difference between PCA-GLUE and GLUE is in estimating the initial displacement image, where GLUE uses DP to estimate the displacement of every single RF line, whereas PCA-GLUE applies DP for only 5 RF lines, then uses a linear combination of previously computed principal components as an initial displacement image. Therefore, if DP fails in estimating the correct displacement for a certain RF line, that means that GLUE would have an incorrect initial displacement image, which affects the fine-tuned displacement image.

The reason behind this robustness is that PCA-GLUE relies on the principal components



(a) B-mode (b) Strain from Skip 1 method

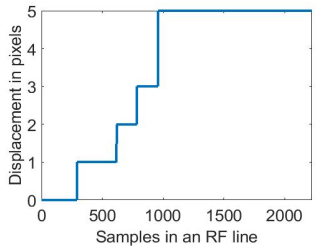
(c) Strain from Skip 2 method (d) Strain from our method

(e) Color bar

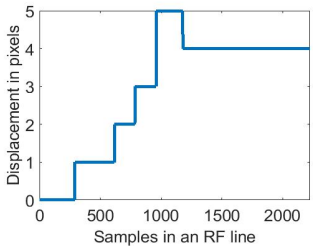
Figure 2.9: The B-mode ultrasound and PCA-GLUE axial strain image for the *in vivo* liver data using different frame selection methods. Note that the pair of RF data used for estimating strain is different from that of Fig. 2.7. The color bar is for the strain images only.

previously computed offline, such that the resulting initial displacement image is represented as a linear combination of them. Therefore, if incorrect results were among the 5 RF lines chosen by PCA-GLUE, it would still be able to estimate the strain correctly due to the additional step of estimating TDE as a sum of principal components.

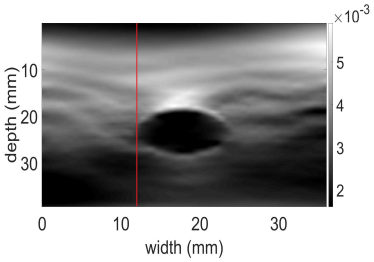
Fig. 2.10 shows how both GLUE and PCA-GLUE perform when they get incorrect initial estimates from DP. Fig. 2.11 shows a comparison between the strain estimated by both GLUE and PCA-GLUE on the finite element method (FEM) simulation data before and after adding a gaussian noise with $\mu = 0$ and $\sigma^2 = 0.1225$ to 10% of the RF lines. The large error on these RF lines could be caused in real life due to air bubbles between the probe and tissue, or large out-of-plane motion in some regions.



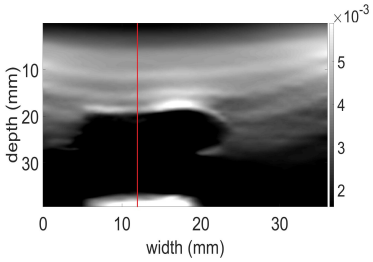
(a) The correct displacement for a certain RF line



(b) The incorrect displacement for a certain RF line



(c) Strain estimated by GLUE



(d) Strain estimated by PCA-GLUE

Figure 2.10: Strain estimated by both GLUE and PCA-GLUE given that DP failed in computing correct initial estimates. The failure occurred in the RF line shown in red.

2.3.4 Varying the number of sparse features

Fig. 2.12 shows the effect of running DP on more than 5 RF lines. We can conclude that the accuracy of the strain estimation does not improve any further when setting p to a value more than 5. As more RF lines correspond to more features and consequently more computations, we choose the smallest value $p = 5$ without sacrificing the accuracy. For more analysis and results at different values of p, please refer to Appendix A of the thesis.

2.4 Discussion

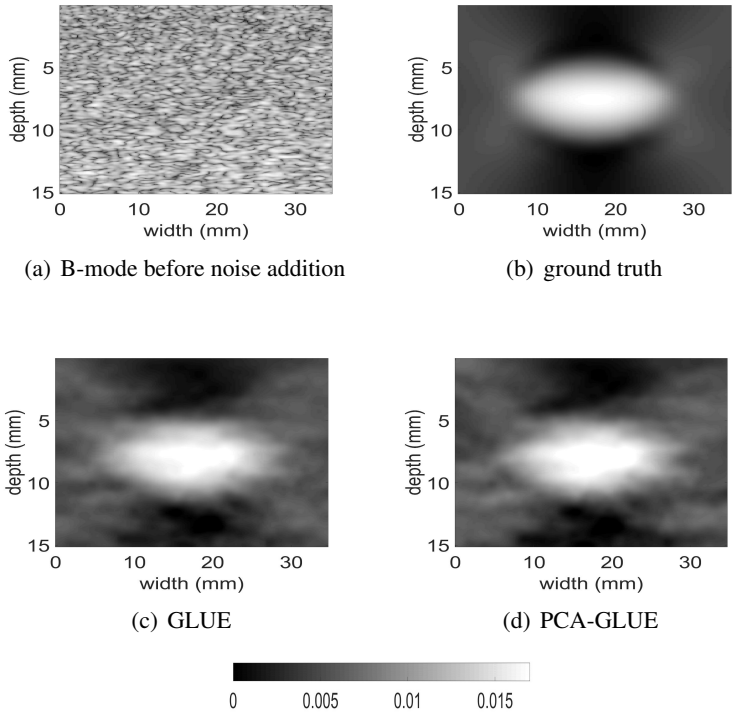
We presented a novel method that can estimate a coarse displacement map from a sparse set of displacement data provided by DP. For an image of size 2304×384, DP takes 163 *ms* and estimation of coarse displacement field takes 95 *ms*, for a total of 258 *ms*. We also presented a novel method

for frame selection that classifies a pair of RF Data as suitable or unsuitable for elastography in only 1.5 *ms*. The input to our classifier is the **w** vector and not the RF data or the displacement image. The reason is that inference with such low-dimensional input is very computationally efficient.

It is worth mentioning that we were only concerned with the axial displacement, as we couldn't compute any principal components that describe the lateral displacement. The reason is that by following Algorithm 3 for the lateral displacement, we found that the variance is not concentrated in the first few eigenvectors, unlike the axial displacement. It is rather almost equally distributed over hundreds of eigenvectors (resembling white noise). We conclude that capturing 95% of the variance would require us to save hundreds of principal components, which is not practical. Therefore, we only use the integer estimates for the $k = m \times p$ pixels computed by DP, followed by bi-linear interpolation which provides an acceptable initial lateral displacement, compared to the alternative approach where we run DP on all RF lines. A comparison between the lateral displacement estimated by the two approaches is shown in Appendix A. The combination of $N = 12$ and $p = 5$ is not a fixed choice of the hyperparameters, as different datasets would require different tuning. In our case, this choice is adequate for all the datasets used.

2.5 Conclusion

In this chapter, we introduced a new method with two main contributions which are fast strain estimation and RF frame selection. In addition, our method is robust to incorrect initial estimates by DP. Our method is more than 10 times faster than GLUE in estimating the initial displacement image, which is the step prior to the exact displacement estimation, while giving the same or better results. Our MLP classifier used for frame selection has been tested on 1,430 unseen pairs of RF frames from both phantom and *in vivo* datasets, and the F1-measure obtained was always higher than 92%. This proves that our method is efficient and that it could be used commercially.



(a) B-mode before noise addition

(b) ground truth

(c) GLUE

(d) PCA-GLUE

(e) Color bar

(f) B-mode after noise addition

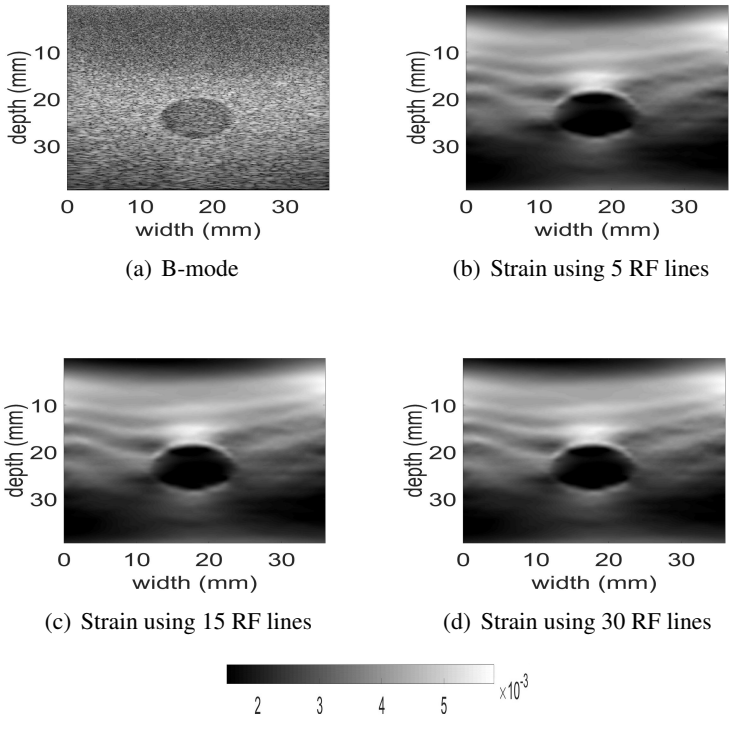
(g) ground truth

(h) GLUE

(i) PCA-GLUE

(j) Color bar

Figure 2.11: The B-mode ultrasound and ground truth axial strain as well as the result of both GLUE and PCA-GLUE for the simulation data before and after adding gaussian noise with $\mu = 0$ and $\sigma^2 = 0.1225$ to 10% of the RF lines. The color bar is for the strain images only.



(a) B-mode

(b) Strain using 5 RF lines

(c) Strain using 15 RF lines

(d) Strain using 30 RF lines

(e) Color bar

Figure 2.12: The B-mode ultrasound and axial strain image using PCA-GLUE for the real phantom experiment as we increase the number of RF lines p from 5 to 30. The color bar is for the strain images only.

Chapter 3

Automatic Frame Selection using CNN in Ultrasound Elastography

As mentioned before in the previous chapter, ultrasound elastography is used to estimate the mechanical properties of the tissue by monitoring its response to an internal or external force. Different levels of deformation are obtained from different tissue types depending on their mechanical properties, where stiffer tissues deform less. Given two radio frequency (RF) frames collected before and after some deformation, we estimate displacement and strain images by comparing the RF frames. The quality of the strain image is dependent on the type of motion that occurs during deformation. In-plane axial motion results in high-quality strain images, whereas out-of-plane motion results in low-quality strain images. In this chapter, we introduce a new method using a convolutional neural network (CNN) to determine the suitability of a pair of RF frames for elastography in only 5.4 *ms*. Our method could also be used to automatically choose the best pair of RF frames, yielding a high-quality strain image. The CNN was trained on 3,818 pairs of RF frames, while testing was done on 986 new unseen pairs, achieving an accuracy of more than 91%. The RF frames were collected from both phantom and *in vivo* data.

3.1 Introduction

As we discussed in Chapter 2, one of the problems that free-hand ultrasound elastography faces is the difficulty in choosing suitable RF frames to estimate the strain. If the two RF frames are collected from the same plane and the force is purely axial, they will yield a high-quality strain image. Therefore, the operator needs to be an expert in performing the freehand palpation, rendering this technique very user-dependent.

In this chapter, we introduce a novel method using a CNN to determine whether a specific pair of RF frames is suitable for elastography. Although we focus on quasi-static elastography, the method can also be applied to other types of elastography. Our MLP model introduced in Chapter 2 relies on a 12-dimentional feature vector to predict the suitability of various RF frame pairs. Using a CNN-based model, we achieve better results by extracting more features, which is possible by considering our input to be the 2 RF frames, rather than the weight vector.

3.2 Methods

In this section, we will discuss data collection for training and testing, and the CNN architecture used. Our model is simply a binary classifier, which is used to determine the suitability of a pair of RF frames for strain estimation.

Our proposed technique can also be used for automatically finding the best RF frames for a specific pre-selected RF frame. The model achieves that by searching in a window composed of several RF frames (in this work, 8 before and after the pre-specified RF frame).

3.2.1 Data Collection

The data used for training and testing the algorithm includes both phantom and *in vivo* data. For the phantom data used in this work, 4,116 pairs of RF frames were collected at Concordia University's PERFORM Centre from 3 different CIRS phantoms (Norfolk, VA), namely Models 040GSE, 039 and 059 at different locations. 3,290 pairs out of the total data were used for training and validation with a ratio of 80:20, and the remaining data was used for testing. The ultrasound device used was the 12R Alpinion ultrasound machine (Bothell, WA) with an L3-12H high density linear array probe at a center frequency of 8.5 MHz and sampling frequency of 40 MHz. For the *in vivo* data, 688 pairs of RF frames were collected at Johns Hopkins Hospital from different patients who were undergoing liver ablation for primary or secondary liver cancers. Detailed information about this data is available in [52]. 528 pairs out of the 688 pairs were used for training and validation with a ratio of 80:20, leaving the rest of the pairs for testing. The labelling of the data was done as described in Algorithm 1.

Algorithm 4 Labelling the dataset for the CNN classifier

1: **procedure**
2: RF frames I_1 and I_2 are passed to PCA-GLUE [51, 82] to obtain the displacement image.
3: I_2 is deformed and interpolated according to the computed displacement image yielding $I_2{}'$.
4: We partition I_1 and $I_2{}'$ into 9 windows.
5: Normalized Cross Correlation (NCC) is calculated between every window in I_1 and its corresponding window in $I_2{}'$, resulting in 9 different NCCs.
6: The final decision is 1 if *both* the smallest NCC is higher than 0.9 *and* the absolute value of the average displacement is more than 0.5 pixels, and 0 otherwise.
7: **end procedure**

It is important to note that steps 2 and 3 in Algorithm 1 are very computationally complex. As such, they cannot be performed in real-time for selecting optimal pairs of RF data. Our proposed method only performs these steps during training, and encodes the results into a computationally efficient CNN.

3.2.2 Architecture

Suppose we have two RF frames I_1 and I_2, and we would like to determine the suitability of this pair for strain estimation. We simply input the two frames to the CNN classifier on two different channels, and the output is a binary number 1 or 0. The architecture used is relatively simple as shown Fig. 3.1. Every convolutional layer has a Rectified Linear Unit (ReLU) as the activation function, and is followed by batch normalization. The activation function in the output layer is a softmax, where the output values in the two nodes represent the probability of having a good and a bad pair respectively. The applied optimization technique is the Adam optimizer [76] with a learning rate of 10^{-3} and a cross entropy loss function. The CNN code is written in Python using Keras.

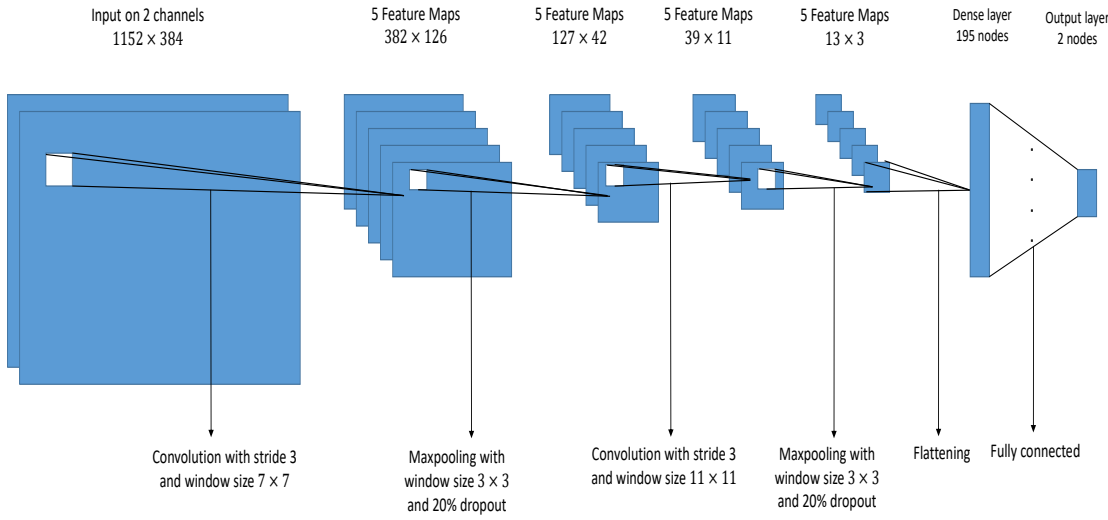


Figure 3.1: The architecture of the CNN used for RF frame selection.

3.2.3 Training and testing time

The labelling of the data, which includes applying Algorithm 1 on every single pair of RF frames took 22 hours. Most of this time was spent on displacement estimation (step 2) and interpolating RF data (step 3). The actual training of the CNN took 7.4 minutes on a 7th generation 3.4 GHz Intel

core i5 desktop with a NVIDIA TITAN V GPU. Inference is very fast, and only takes 5.4 *ms* to classify two frames of size 2304 by 384. The frames are downsampled by a factor of 2 in the axial direction, to generate smaller input images for the CNN. Note that in comparison, doing steps 2, 3, 5 and 6 in Algorithm 1 for two frames of the same size takes 6.21 seconds, 14.04 seconds, 46.87 *ms* and 2.45 *ms* respectively, for a total run-time of 20.3 seconds. In other words, frame selection with CNN is more than 3,700 times faster. It is important to note than CNN computations are performed on a GPU, whereas the steps in Algorithm 1 use a CPU.

3.3 Results

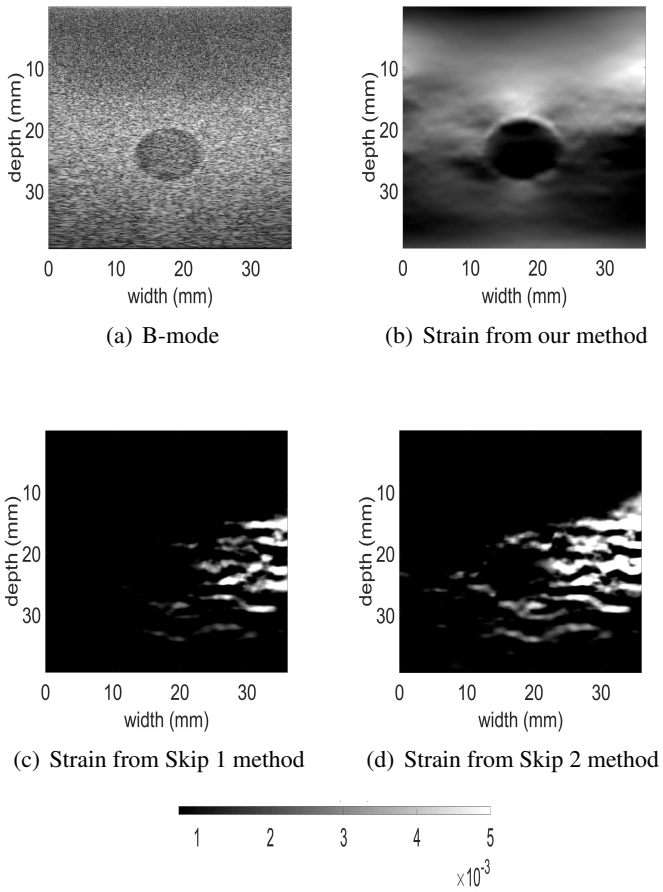
In this section, we compare our CNN frame selection method to other methods that choose to pair an RF frame with another by simply skipping one or two frames.

Fig. 3.2 shows the output of different frame selection methods when tested on one of the phantom datasets. It is clear that our automatic frame selection substantially outperforms the fixed skip frame pairing methods as it chooses more suitable frames, yielding better quality strain images. Table 3.1 shows the accuracy as well as the F1-measure obtained from our CNN classifier on new phantom datasets, that were not used during training. The results prove the ability of the classifier to generalize to unseen data.

Table 3.1: The accuracy and F1-measure of our CNN classifier on the phantom and *in vivo* test data.

Dataset	**Size**	**Accuracy**	**F1-measure**
Phantom dataset 1	228 instances	96.77%	93.68%
Phantom dataset 2	297 instances	91.7%	89.17%
Phantom dataset 3	301 instances	96%	96%
In vivo dataset	160 instances	95.24%	92%

Fig. 3.3 shows a comparison between the performance of our method and the fixed skip frame



(a) B-mode (b) Strain from our method

(c) Strain from Skip 1 method (d) Strain from Skip 2 method

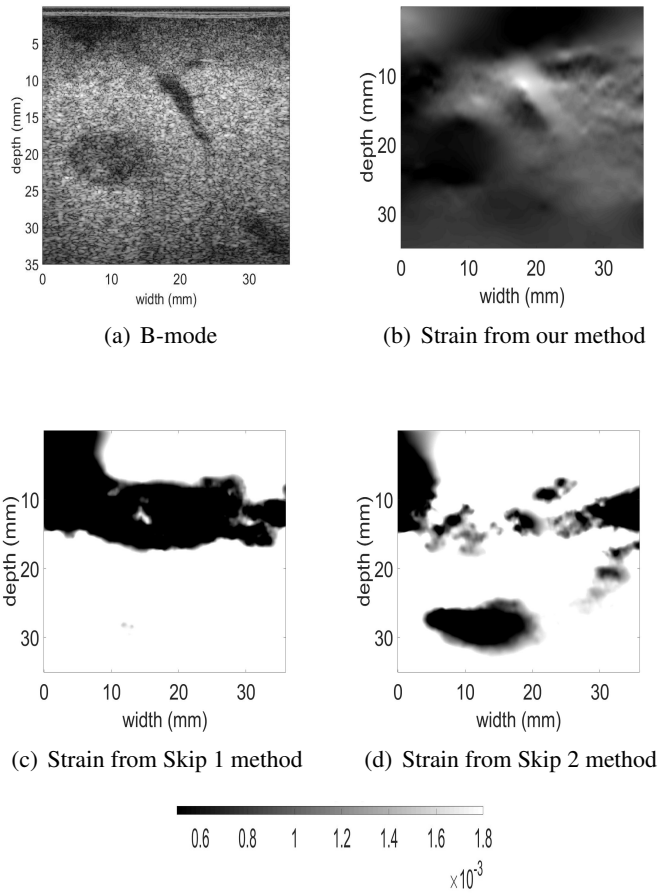
(e) Color bar

Figure 3.2: The B-mode ultrasound and PCA-GLUE axial strain image for the phantom experiment using different frame selection methods. Poor strain images in (c) and (d) are rejected by the proposed method. The color bar is for the strain images.

pairing on the *in vivo* dataset. Table 3.1 shows the accuracy as well as the F1-measure obtained from our method. Again, it is clear that our CNN-based method performs substantially better.

3.4 Conclusion

In this chapter we introduced a new method based on CNN to automatically choose RF frames that are suitable for strain estimation. Our method is fast, practical and does not need any external hardware. Therefore, it could be used commercially to generate high quality strain images even



(a) B-mode (b) Strain from our method

(c) Strain from Skip 1 method (d) Strain from Skip 2 method

(e) Color bar

Figure 3.3: The B-mode ultrasound and PCA-GLUE axial strain image for the *in vivo* experiment using different frame selection methods. The low-quality strain images in (c) and (d) are rejected by the proposed method. The color bar is for the strain images.

when used by an inexperienced operator. This can be achieved by simply giving a warning message to the sonographer if the frames used are not suitable.

Chapter 4

Conclusions and Future Work

4.1 Conclusions

In this thesis, we introduced three different techniques based on machine learning to solve the problems of fast strain estimation and RF frame selection in ultrasound elastography. Our models were trained on data from both phantom and *in vivo* datasets. During testing, the 3 models were able to generalize with high accuracy to unseen data and to outperform other already existing methods that target the same end goal. Our first model, PCA-GLUE, uses Principal Components Analysis (PCA) to estimate the initial displacement image from two RF frames more than 10 times faster than GLUE, while giving the same results on a wide range of datasets. In addition, PCA-GLUE outperforms other similar methods such as NCC. We also showed that it is more robust to high levels of noise in RF data compared to GLUE. Having such a fast and robust method for strain estimation is of high importance in clinical translation.

Our second model is based on an MLP classifier to distinguish suitable from unsuitable RF frame pairs for elastography. It can also be used to search for the best RF frame to be paired with a given pre-selected one. By comparing our method to other methods such as fixed skip frame pairing

or methods that use a 3D camera [66], an external tracker for selecting the RF frames [64, 83] or a mechanical device for forcing the motion to be in-plane [62], we can conclude that our method is the only one that can successfully choose suitable RF frames without needing any external device, making it easy-to-use and versatile.

In our third model, we improved the MLP classifier by using a CNN-based network that can automatically extract more features. Our conclusion is that having more features substantially improves the accuracy of automatic RF frame selection. The only drawback is that we need to have a GPU for training the CNN model, but this is done once. Inference takes only 5.4 *ms* and does not need a GPU.

4.2 Future Work

Our work can be extended in different ways. On one hand, our strain estimation that relies on PCA to decompose the initial displacement image into a linear combination of principal components can be improved by using an autoencoder to project displacement images into a low-dimensional space [84]. This will allow us to use more principal components while projecting them into a lower-dimensional space to reduce the storage. During testing, we can project them back to their original space to be used for displacement estimation. This would reduce storage space and improve the accuracy of displacement estimation and consequently strain estimation. Autoencoders, however, need more data to be trained efficiently, which means we would need more data that capture different deformations and collected by different users from several ultrasound devices. Furthermore, investigating how the compactional complexity scales with the size of RF frames is also very important, as it is related to the speed of our method, which is considered one of its main advantages.

On the other hand, our RF frame selection model could be improved by having a larger dataset

and trying different CNN architectures to choose the best one based on the performance on the validation dataset. Doing an ablation study will also help us in choosing the best architecture by measuring the performance after the removal of different parts of the network. Since transfer learning is considered more efficient than learning from scratch [85], we can use an already existing CNN architecture and fine-tune it for our task. Some examples of popular CNN architectures include: AlexNet [86], GoogLeNet [87] and ResNet [88]. This could potentially slightly improve the accuracy of our CNN model, which is currently between 92% and 97% depending on the dataset. In addition, data augmentation could be applied to the training dataset used by the CNN classifier. Data augmentation is mainly done to introduce some variability in the training set to make the classifier more robust. We simply rotate the RF frame pairs by different degrees (for example 30 degrees) both clockwise and anticlockwise, then feed the resulting images to the classifier. Moreover, the task of RF frame selection could be implemented by various classifiers such as logistic regression, Gaussian discriminant analysis, quadratic discriminant analysis, naive Bayes classifier, random forest or support vector machine. Doing a comparison between the performance of these classifiers would help us to find the most adequate model for our task. More future work can also include handling imbalanced datasets using various methods such as [89–91]. We relied on a simple approach to tackle this problem by sampling from the overrepresented class to balance the dataset. More performance metrics can also be used to assess the quality of our classifier such as the area under the receiver operating characteristic (ROC) curve (AUC), which measures the area under the graph showing the relation between the true positive rate and false positive rate.

In order to extract the principal components used by PCA-GLUE, we used GLUE to calculate the displacement of 3,772 pairs of RF frames. Better results would be obtained if we replace GLUE by a better displacement estimation method such as Global Ultrasound Elastography in

Spatial and Temporal Domains (GUEST) [53] or tOtal Variation Regularization and WINDow-based time delay estimation (OVERWIND) [92]. OVERWIND combines both window-based and optimization-based methods for displacement estimation, while using the L1 as the regularization term of the cost function. Therefore, OVERWIND is more robust to signal decorrelation and yields sharper displacement images compared to GLUE. As for GUEST, it uses three RF frames to form a cost function that is regularized in both the temporal and spatial domains, thus outperforming GLUE which doesn't make use of the temporal information. Finally, the principal components learned by PCA-GLUE can be used to provide fast initial displacement estimation for various strain estimation methods other than GLUE, such as GUEST.

References

[1] M. Ashikuzzaman, C. Belasso, M. G. Kibria, A. Bergdahl, C. J. Gauthier, and H. Rivaz, "Low rank and sparse decomposition of ultrasound color flow images for suppressing clutter in real-time," *IEEE transactions on medical imaging*, vol. 39, no. 4, pp. 1073–1084, 2020.

[2] A. Kornecki, "Current status of breast ultrasound," *Canadian Association of Radiologists' Journal*, vol. 62, no. 1, pp. 31–40, 2011.

[3] J. A. Cruz and D. S. Wishart, "Applications of machine learning in cancer prediction and prognosis," *Cancer informatics*, vol. 2, p. 117693510600200030, 2006.

[4] V. Vishrutha and M. Ravishankar, "Early detection and classification of breast cancer," in *Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014*, pp. 413–419, Springer, 2015.

[5] M. M. R. Krishnan, S. Banerjee, C. Chakraborty, C. Chakraborty, and A. K. Ray, "Statistical analysis of mammographic features and its classification using support vector machine," *Expert Systems with Applications*, vol. 37, no. 1, pp. 470–478, 2010.

[6] J.-Z. Cheng, D. Ni, Y.-H. Chou, J. Qin, C.-M. Tiu, Y.-C. Chang, C.-S. Huang, D. Shen, and C.-M. Chen, "Computer-aided diagnosis with deep learning architecture: applications to breast

lesions in us images and pulmonary nodules in ct scans," *Scientific reports*, vol. 6, no. 1, pp. 1–13, 2016.

[7] A. Jalalian, S. B. Mashohor, H. R. Mahmud, M. I. B. Saripan, A. R. B. Ramli, and B. Karasfi, "Computer-aided detection/diagnosis of breast cancer in mammography and ultrasound: a review," *Clinical imaging*, vol. 37, no. 3, pp. 420–426, 2013.

[8] S. Han, H.-K. Kang, J.-Y. Jeong, M.-H. Park, W. Kim, W.-C. Bang, and Y.-K. Seong, "A deep learning framework for supporting the classification of breast lesions in ultrasound images," *Physics in Medicine & Biology*, vol. 62, no. 19, p. 7714, 2017.

[9] K. Doi, "Computer-aided diagnosis in medical imaging: historical review, current status and future potential," *Computerized medical imaging and graphics*, vol. 31, no. 4-5, pp. 198–211, 2007.

[10] B. Van Ginneken, C. M. Schaefer-Prokop, and M. Prokop, "Computer-aided diagnosis: how to move from the laboratory to the clinic," *Radiology*, vol. 261, no. 3, pp. 719–732, 2011.

[11] M. L. Giger, H.-P. Chan, and J. Boone, "Anniversary paper: history and status of cad and quantitative image analysis: the role of medical physics and aapm," *Medical physics*, vol. 35, no. 12, pp. 5799–5820, 2008.

[12] J.-Z. Cheng, Y.-H. Chou, C.-S. Huang, Y.-C. Chang, C.-M. Tiu, K.-W. Chen, and C.-M. Chen, "Computer-aided us diagnosis of breast lesions by using cell-based contour grouping," *Radiology*, vol. 255, no. 3, pp. 746–754, 2010.

[13] M. L. Giger, N. Karssemeijer, and J. A. Schnabel, "Breast image analysis for risk assessment, detection, diagnosis, and treatment of cancer," *Annual review of biomedical engineering*, vol. 15, pp. 327–357, 2013.

[14] S. Joo, Y. S. Yang, W. K. Moon, and H. C. Kim, "Computer-aided diagnosis of solid breast nodules: use of an artificial neural network based on multiple sonographic features," *IEEE transactions on medical imaging*, vol. 23, no. 10, pp. 1292–1300, 2004.

[15] C.-M. Chen, Y.-H. Chou, K.-C. Han, G.-S. Hung, C.-M. Tiu, H.-J. Chiou, and S.-Y. Chiou, "Breast lesions on sonograms: computer-aided diagnosis with nearly setting-independent features and artificial neural networks," *Radiology*, vol. 226, no. 2, pp. 504–514, 2003.

[16] K. Drukker, C. A. Sennett, and M. L. Giger, "Automated method for improving system performance of computer-aided diagnosis in breast ultrasound," *IEEE Transactions on medical imaging*, vol. 28, no. 1, pp. 122–128, 2008.

[17] K. Awai, K. Murao, A. Ozawa, Y. Nakayama, T. Nakaura, D. Liu, K. Kawanaka, Y. Funama, S. Morishita, and Y. Yamashita, "Pulmonary nodules: estimation of malignancy at thin-section helical ct—effect of computer-aided diagnosis on performance of radiologists," *Radiology*, vol. 239, no. 1, pp. 276–284, 2006.

[18] M. B. McCarville, H. M. Lederman, V. M. Santana, N. C. Daw, S. J. Shochat, C.-S. Li, and R. A. Kaufman, "Distinguishing benign from malignant pulmonary nodules with helical chest ct in children with malignant solid tumors," *Radiology*, vol. 239, no. 2, pp. 514–520, 2006.

[19] I. C. Sluimer, P. F. van Waes, M. A. Viergever, and B. van Ginneken, "Computer-aided diagnosis in high resolution ct of the lungs," *Medical physics*, vol. 30, no. 12, pp. 3081–3090, 2003.

[20] T. Sun, R. Zhang, J. Wang, X. Li, and X. Guo, "Computer-aided diagnosis for early-stage lung cancer based on longitudinal and balanced data," *PloS one*, vol. 8, no. 5, 2013.

[21] T. W. Way, B. Sahiner, H.-P. Chan, L. Hadjiiski, P. N. Cascade, A. Chughtai, N. Bogot, and E. Kazerooni, "Computer-aided diagnosis of pulmonary nodules on ct scans: improvement of classification performance with nodule surface features," *Medical physics*, vol. 36, no. 7, pp. 3086–3098, 2009.

[22] S. G. Armato III and W. F. Sensakovic, "Automated lung segmentation for thoracic ct: Impact on computer-aided diagnosis1," *Academic Radiology*, vol. 11, no. 9, pp. 1011–1021, 2004.

[23] T. W. Way, L. M. Hadjiiski, B. Sahiner, H.-P. Chan, P. N. Cascade, E. A. Kazerooni, N. Bogot, and C. Zhou, "Computer-aided diagnosis of pulmonary nodules on ct scans: Segmentation and classification using 3d active contours," *Medical physics*, vol. 33, no. 7Part1, pp. 2323–2337, 2006.

[24] J. Wang, X. Yang, H. Cai, W. Tan, C. Jin, and L. Li, "Discrimination of breast cancer with microcalcifications on mammography by deep learning," *Scientific reports*, vol. 6, no. 1, pp. 1–9, 2016.

[25] B. Behboodi and H. Rivaz, "Ultrasound segmentation using u-net: learning from simulated data and testing on real data," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 6628–6631, IEEE, 2019.

[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[27] S. Goudarzi, A. Asif, and H. Rivaz, "Multi-focus ultrasound imaging using generative adversarial networks," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pp. 1118–1121, IEEE, 2019.

[28] T. J Hall, P. E Barboneg, A. A Oberai, J. Jiang, J.-F. Dord, S. Goenezen, and T. G Fisher, "Recent results in nonlinear strain and modulus imaging," *Current medical imaging reviews*, vol. 7, no. 4, pp. 313–327, 2011.

[29] J.-L. Gennisson, T. Deffieux, M. Fink, and M. Tanter, "Ultrasound elastography: principles and techniques," *Diagnostic and interventional imaging*, vol. 94, no. 5, pp. 487–495, 2013.

[30] R. M. Sigrist, J. Liau, A. El Kaffas, M. C. Chammas, and J. K. Willmann, "Ultrasound elastography: review of techniques and clinical applications," *Theranostics*, vol. 7, no. 5, p. 1303, 2017.

[31] K. Fujimoto, M. Kato, M. Kudo, N. Yada, T. Shiina, K. Ueshima, Y. Yamada, T. Ishida, M. Azuma, M. Yamasaki, *et al.*, "Novel image analysis method using ultrasound elastography for noninvasive evaluation of hepatic fibrosis in patients with chronic hepatitis c," *Oncology*, vol. 84, no. Suppl. 1, pp. 3–12, 2013.

[32] M. Menzilcioglu, M. Duymus, S. Citil, S. Avcu, G. Gungor, T. Sahin, S. Boysan, O. Altunoren, and A. Sarica, "Strain wave elastography for evaluation of renal parenchyma in chronic kidney disease," *The British journal of radiology*, vol. 88, no. 1050, p. 20140714, 2015.

[33] R. G. Barr, K. Nakashima, D. Amy, D. Cosgrove, A. Farrokh, F. Schafer, J. C. Bamber, L. Castera, B. I. Choi, Y.-H. Chou, *et al.*, "Wfumb guidelines and recommendations for clinical use of ultrasound elastography: Part 2: breast," *Ultrasound in medicine & biology*, vol. 41, no. 5, pp. 1148–1160, 2015.

[34] M. Brock, C. von Bodman, R. J. Palisaar, B. Löppenberg, F. Sommerer, T. Deix, J. Noldus, and T. Eggert, "The impact of real-time elastography guiding a systematic prostate biopsy to

improve cancer detection rate: a prospective study of 353 patients," *The Journal of urology*, vol. 187, no. 6, pp. 2039–2043, 2012.

[35] J. Bojunga, E. Herrmann, G. Meyer, S. Weber, S. Zeuzem, and M. Friedrich-Rust, "Real-time elastography for the differentiation of benign and malignant thyroid nodules: a meta-analysis," *Thyroid*, vol. 20, no. 10, pp. 1145–1150, 2010.

[36] G. Ferraioli, C. Filice, L. Castera, B. I. Choi, I. Sporea, S. R. Wilson, D. Cosgrove, C. F. Dietrich, D. Amy, J. C. Bamber, *et al.*, "Wfumb guidelines and recommendations for clinical use of ultrasound elastography: Part 3: liver," *Ultrasound in medicine & biology*, vol. 41, no. 5, pp. 1161–1179, 2015.

[37] K. Parker, M. Doyley, and D. Rubens, "Imaging the elastic properties of tissue: the 20 year perspective," *Physics in medicine & biology*, vol. 56, no. 1, p. R1, 2010.

[38] J. Ophir, I. Cespedes, H. Ponnekanti, Y. Yazdi, and X. Li, "Elastography: a quantitative method for imaging the elasticity of biological tissues," *Ultrasonic imaging*, vol. 13, no. 2, pp. 111–134, 1991.

[39] Q. He, L. Tong, L. Huang, J. Liu, Y. Chen, and J. Luo, "Performance optimization of lateral displacement estimation with spatial angular compounding," *Ultrasonics*, vol. 73, pp. 9–21, 2017.

[40] Y. Wang, M. Bayer, J. Jiang, and T. J. Hall, "Large-strain 3-d in vivo breast ultrasound strain elastography using a multi-compression strategy and a whole-breast scanning system," *Ultrasound in medicine & biology*, vol. 45, no. 12, pp. 3145–3159, 2019.

[41] R. M. Pohlman and T. Varghese, "Physiological motion reduction using lagrangian tracking for electrode displacement elastography," *Ultrasound in Medicine & Biology*, 2019.

[42] G.-Y. Li, Q. He, G. Xu, L. Jia, J. Luo, and Y. Cao, "An ultrasound elastography method to determine the local stiffness of arteries with guided circumferential waves," *Journal of biomechanics*, vol. 51, pp. 97–104, 2017.

[43] S. Wu, Z. Gao, Z. Liu, J. Luo, H. Zhang, and S. Li, "Direct reconstruction of ultrasound elastography using an end-to-end deep neural network," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 374–382, Springer, 2018.

[44] C. Hoerig, J. Ghaboussi, and M. F. Insana, "Data-driven elasticity imaging using cartesian neural network constitutive models and the autoprogressive method," *IEEE transactions on medical imaging*, vol. 38, no. 5, pp. 1150–1160, 2018.

[45] B. Peng, Y. Xian, and J. Jiang, "A convolution neural network-based speckle tracking method for ultrasound elastography," in *2018 IEEE International Ultrasonics Symposium (IUS)*, pp. 206–212, IEEE, 2018.

[46] Z. Gao, S. Wu, Z. Liu, J. Luo, H. Zhang, M. Gong, and S. Li, "Learning the implicit strain reconstruction in ultrasound elastography using privileged information," *Medical image analysis*, vol. 58, p. 101534, 2019.

[47] M. G. Kibria and H. Rivaz, "Gluenet: Ultrasound elastography using convolutional neural network," in *Simulation, Image Processing, and Ultrasound Systems for Assisted Diagnosis and Navigation*, pp. 21–28, Springer, 2018.

[48] B. Peng, Y. Xian, Q. Zhang, and J. Jiang, "Neural network-based motion tracking for breast ultrasound strain elastography: An initial assessment of performance and feasibility," *Ultrasonic Imaging*, p. 0161734620902527, 2020.

[49] A. K. Tehrani and H. Rivaz, "Displacement estimation in ultrasound elastography using pyramidal convolutional neural network," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 2020.

[50] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8934–8943, 2018.

[51] H. S. Hashemi and H. Rivaz, "Global time-delay estimation in ultrasound elastography," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 64, no. 10, pp. 1625–1636, 2017.

[52] H. Rivaz, E. M. Boctor, M. A. Choti, and G. D. Hager, "Real-time regularized ultrasound elastography," *IEEE transactions on medical imaging*, vol. 30, no. 4, pp. 928–945, 2011.

[53] M. Ashikuzzaman, C. J. Gauthier, and H. Rivaz, "Global ultrasound elastography in spatial and temporal domains," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 2019.

[54] J. Jiang and T. J. Hall, "A coupled subsample displacement estimation method for ultrasound-based strain elastography," *Physics in Medicine & Biology*, vol. 60, no. 21, p. 8347, 2015.

[55] L. Yuan and P. C. Pedersen, "Analytical phase-tracking-based strain estimation for ultrasound elasticity," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 62, no. 1, pp. 185–207, 2015.

[56] J. Luo and E. E. Konofagou, "A fast normalized cross-correlation calculation method for motion estimation," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 57, no. 6, pp. 1347–1357, 2010.

[57] J. Wulff and M. J. Black, "Efficient sparse-to-dense optical flow estimation using a learned basis and layers," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) 2015*, June 2015.

[58] R. M. Pohlman and T. Varghese, "Dictionary representations for electrode displacement elastography," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 65, no. 12, pp. 2381–2389, 2018.

[59] T. J. Hall, Y. Zhu, and C. S. Spalding, "In vivo real-time freehand palpation imaging," *Ultrasound in medicine & biology*, vol. 29, no. 3, pp. 427–435, 2003.

[60] R. Chandrasekhar, J. Ophir, T. Krouskop, and K. Ophir, "Elastographic image quality vs. tissue motion in vivo," *Ultrasound in medicine & biology*, vol. 32, no. 6, pp. 847–855, 2006.

[61] M. A. Lubinski, S. Y. Emelianov, and M. O'Donnell, "Adaptive strain estimation using retrospective processing [medical us elasticity imaging]," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 46, no. 1, pp. 97–107, 1999.

[62] K. M. Hiltawsky, M. Krüger, C. Starke, L. Heuser, H. Ermert, and A. Jensen, "Freehand ultrasound elastography of breast lesions: clinical results," *Ultrasound in medicine & biology*, vol. 27, no. 11, pp. 1461–1469, 2001.

[63] J. Jiang, T. J. Hall, and A. M. Sommer, "A novel performance descriptor for ultrasonic strain imaging: A preliminary study," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 53, no. 6, pp. 1088–1102, 2006.

[64] H. Rivaz, P. Foroughi, I. Fleming, R. Zellars, E. Boctor, and G. Hager, "Tracked regularized ultrasound elastography for targeting breast radiotherapy," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 507–515, Springer, 2009.

[65] P. Foroughi, H.-J. Kang, D. A. Carnegie, M. G. van Vledder, M. A. Choti, G. D. Hager, and E. M. Boctor, "A freehand ultrasound elastography system with tracking for in vivo applications," *Ultrasound in medicine & biology*, vol. 39, no. 2, pp. 211–225, 2013.

[66] B. J. Ranger, M. Feigin, X. Zhang, A. Mireault, R. Raskar, H. M. Herr, and B. W. Anthony, "3d optical imagery for motion compensation in a limb ultrasound system," in *Medical Imaging 2016: Ultrasonic Imaging and Tomography*, vol. 9790, p. 97900R, International Society for Optics and Photonics, 2016.

[67] F. Aalamifar, A. Cheng, Y. Kim, X. Hu, H. K. Zhang, X. Guo, and E. M. Boctor, "Robot-assisted automatic ultrasound calibration," *International Journal of Computer Assisted Radiology and Surgery*, vol. 11, pp. 1821–1829, Oct 2016.

[68] A. Zayed and H. Rivaz, "Fast approximate time-delay estimation in ultrasound elastography using principal component analysis," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 6204–6207, IEEE, 2019.

[69] A. Zayed and H. Rivaz, "Automatic frame selection using mlp neural network in ultrasound elastography," in *International Conference on Image Analysis and Recognition*, pp. 462–472, Springer, 2019.

[70] H. Rivaz, E. Boctor, P. Foroughi, R. Zellars, G. Fichtinger, and G. Hager, "Ultrasound elastography: a dynamic programming approach," *IEEE transactions on medical imaging*, vol. 27, no. 10, pp. 1373–1377, 2008.

[71] B. S. Garra, "Elastography: history, principles, and technique comparison," *Abdominal imaging*, vol. 40, no. 4, pp. 680–697, 2015.

[72] B. D. de Vos, F. F. Berendsen, M. A. Viergever, M. Staring, and I. Išgum, "End-to-end unsupervised deformable image registration with a convolutional neural network," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 204–212, Springer, 2017.

[73] S. Chicotay, E. David, and N. S. Netanyahu, "A two-phase genetic algorithm for image registration," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 189–190, 2017.

[74] H. Li and Y. Fan, "Non-rigid image registration using self-supervised fully convolutional networks without training data," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pp. 1075–1078, IEEE, 2018.

[75] A. Subramaniam, P. Balasubramanian, and A. Mittal, "Ncc-net: Normalized cross correlation based deep matcher with robustness to illumination variations," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1944–1953, IEEE, 2018.

[76] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[77] F. Chollet *et al.*, "Keras." `https://keras.io`, 2015.

[78] J. A. Jensen, "Field: A program for simulating ultrasound systems," in *10TH NORDICBALTIC CONFERENCE ON BIOMEDICAL IMAGING, VOL. 4, SUPPLEMENT 1, PART 1: 351–353*, Citeseer, 1996.

[79] J. A. Jensen and N. B. Svendsen, "Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 39, no. 2, pp. 262–267, 1992.

[80] J. Ophir, S. K. Alam, B. Garra, F. Kallel, E. Konofagou, T. Krouskop, and T. Varghese, "Elastography: ultrasonic estimation and imaging of the elastic properties of tissues," *Proceedings Inst. Mech. Eng., Part H: J. Eng. in Med.*, vol. 213, no. 3, pp. 203–233, 1999.

[81] F. Kallel and J. Ophir, "A least-squares strain estimator for elastography," *Ultrasonic imaging*, vol. 19, no. 3, pp. 195–208, 1997.

[82] A. Zayed and H. Rivaz, "Fast approximate time-delay estimation in ultrasound elastography using principal component analysis," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 6204–6207, IEEE, 2019.

[83] P. Foroughi, H. Rivaz, I. N. Fleming, G. D. Hager, and E. M. Boctor, "Tracked ultrasound elastography (true)," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 9–16, Springer, 2010.

[84] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of machine learning research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[85] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[86] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[87] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[88] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[89] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Rusboost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 1, pp. 185–197, 2009.

[90] S. Hido, H. Kashima, and Y. Takahashi, "Roughly balanced bagging for imbalanced data," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 2, no. 5-6, pp. 412–426, 2009.

[91] C. Li, "Classifying imbalanced data using a bagging ensemble variation (bev)," in *Proceedings of the 45th annual southeast regional conference*, pp. 203–208, 2007.

[92] M. Mirzaei, A. Asif, and H. Rivaz, "Combining total variation regularization with window-based time delay estimation in ultrasound elastography," *IEEE transactions on medical imaging*, vol. 38, no. 12, pp. 2744–2754, 2019.

Appendix A

Supplementary Material for

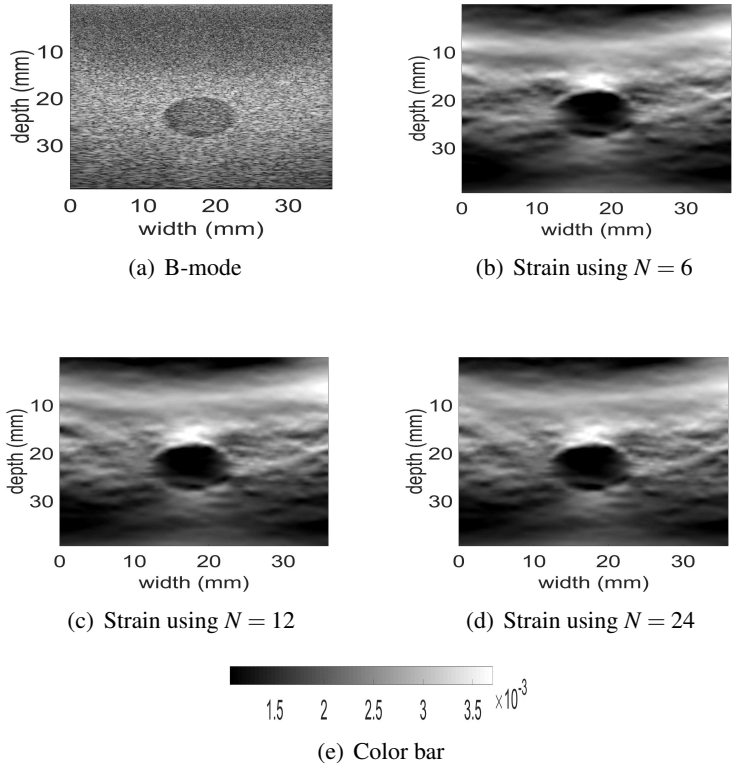
PCA-GLUE

This appendix provides results that are complementary to those in the thesis. We investigate the effect of changing the hyperparameter N, which refers to the number of principal components used, for both phantom and *in vivo* data. We also simulate the effect of changing the hyperparameter p, which refers to the number of radio frequency (RF) lines used for extracting the sparse features. Depending on the rate of change of the displacement image in the horizontal direction, different values of p are needed.

In addition, we use simulation data to show that our method works for different compression levels. Finally, we perform an experiment to show that it is acceptable to only compute the lateral displacement for p RF lines, followed by bi-linear interpolation, as compared to the more computationally expensive solution of calculating the displacement of all RF lines. We compare the axial strain produced by GLobal Ultrasound Elastography (GLUE) while providing it with an initial lateral displacement using both methods.

A.1 Results

We used different number of principal components for both phantom and *in vivo* datasets to justify our choice of $N = 12$. Fig. A.1, A.2 and A.3 show that $N = 6$ is a choice that works for all the datasets except the *in vivo* dataset after ablation. Therefore, we set $N = 12$ for our experiments.



(a) B-mode

(b) Strain using $N = 6$

(c) Strain using $N = 12$

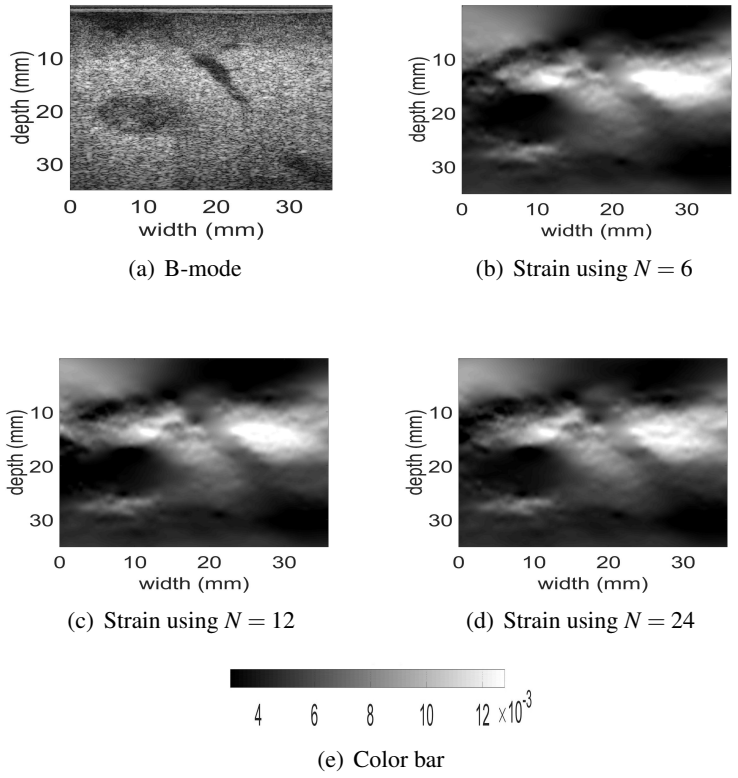
(d) Strain using $N = 24$

(e) Color bar

Figure A.1: The B-mode ultrasound and axial strain image using PCA-GLUE for the real phantom experiment as we increase the number of principal components N from 6 to 24. The color bar is for the strain images only.

Fig. A.4, A.5 and A.6 show that $p = 2$ is also a choice that works for all the datasets except the *in vivo* dataset after ablation. Therefore, we set $p = 5$ since it doesn't noticeably increase the computational complexity.

Tables A.1 and A.2 show a comparison between the signal to Noise Ratio (SNR) and Contrast to Noise Ratio (CNR) of both PCA-GLUE and GLUE for different levels of compression from 1%



(a) B-mode

(b) Strain using $N = 6$

(c) Strain using $N = 12$

(d) Strain using $N = 24$

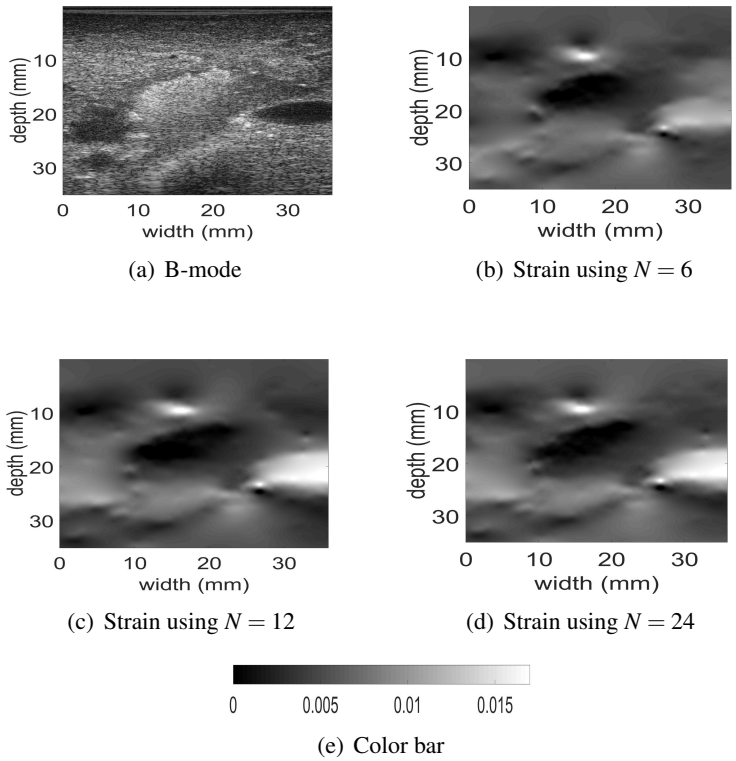
(e) Color bar

Figure A.2: The B-mode ultrasound and axial strain image using PCA-GLUE for the *in vivo* liver data before ablation as we increase the number of principal components N from 6 to 24. The color bar is for the strain images only.

to 6%. We also compare our approach to the more computationally expensive method where the lateral displacement images given to GLUE are obtained by passing on all RF lines. Fig. A.7, A.8 and A.9 show that the 2 methods yield the same results on both phantom and *in vivo* datasets.

Table A.1: The CNR values of the axial strain images for the simulation data at different compression levels.

Compression level	**GLUE**	**PCA-GLUE**
1%	**21.56**	19.74
3%	**21.59**	20.23
6%	**22.06**	19.58



(a) B-mode

(b) Strain using $N = 6$

(c) Strain using $N = 12$

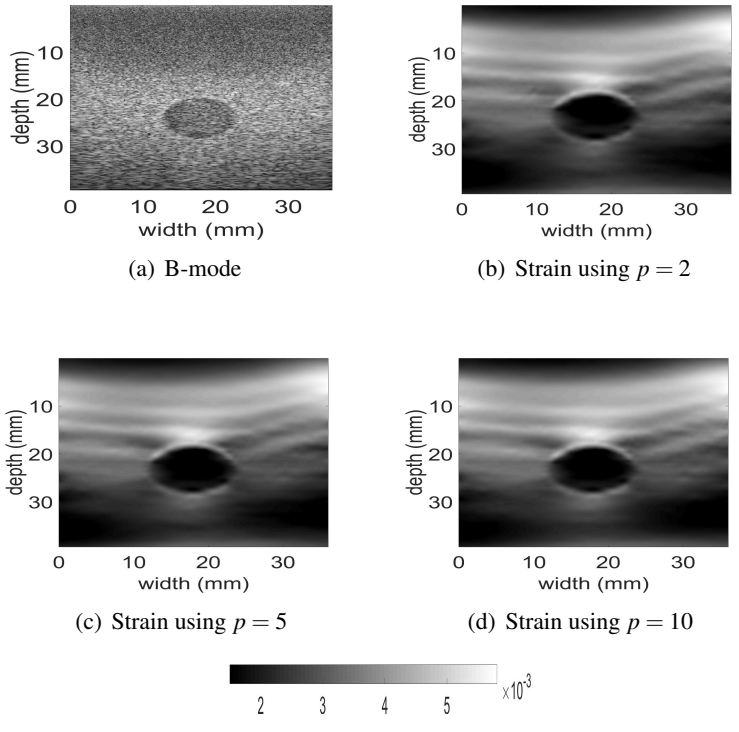
(d) Strain using $N = 24$

(e) Color bar

Figure A.3: The B-mode ultrasound and axial strain image using PCA-GLUE for *in vivo* liver data after ablation as we increase the number of principal components N from 6 to 24. The color bar is for the strain images only.

Table A.2: The SNR values of the axial strain images for the simulation data at different compression levels.

Compression level	**GLUE**	**PCA-GLUE**
1%	22.58	**22.86**
3%	**25.47**	23.53
6%	**28.38**	27.98



(a) B-mode

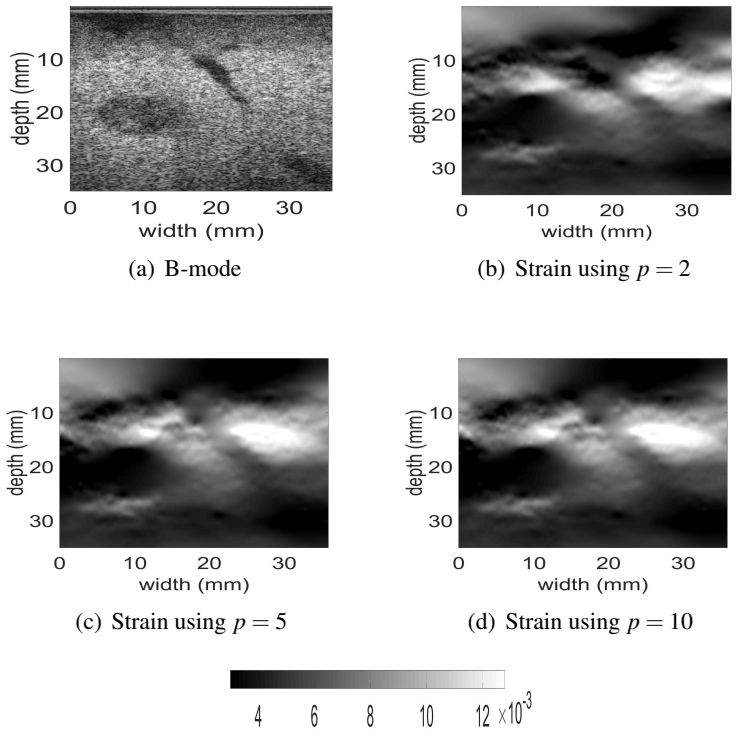
(b) Strain using $p = 2$

(c) Strain using $p = 5$

(d) Strain using $p = 10$

(e) Color bar

Figure A.4: The B-mode ultrasound and axial strain image using PCA-GLUE for the real phantom experiment as we increase the number of RF lines p from 2 to 10. The color bar is for the strain images only.



(a) B-mode

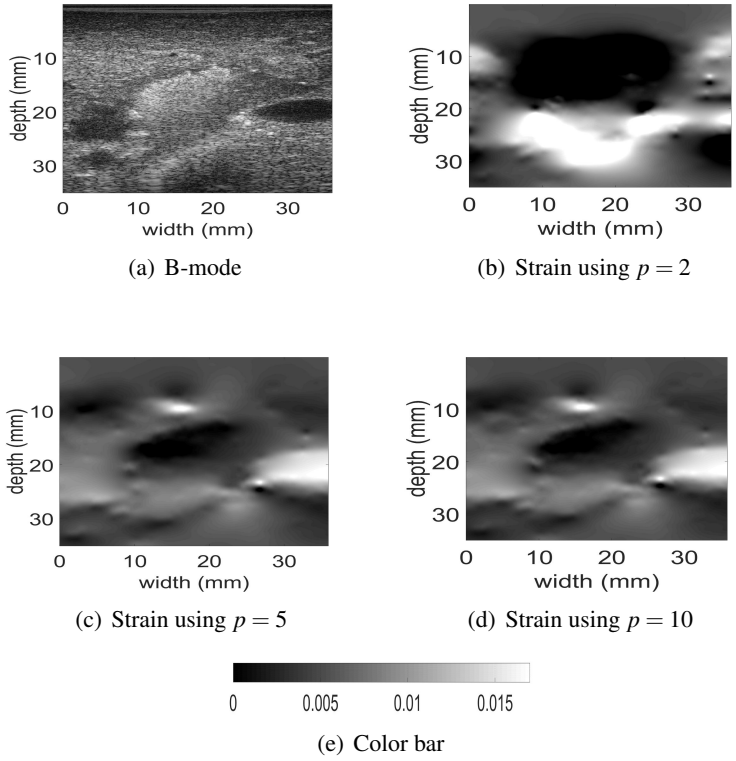
(b) Strain using $p = 2$

(c) Strain using $p = 5$

(d) Strain using $p = 10$

(e) Color bar

Figure A.5: The B-mode ultrasound and axial strain image using PCA-GLUE for the *in vivo* liver data before ablation as we increase the number of RF lines p from 2 to 10. The color bar is for the strain images only.



(a) B-mode

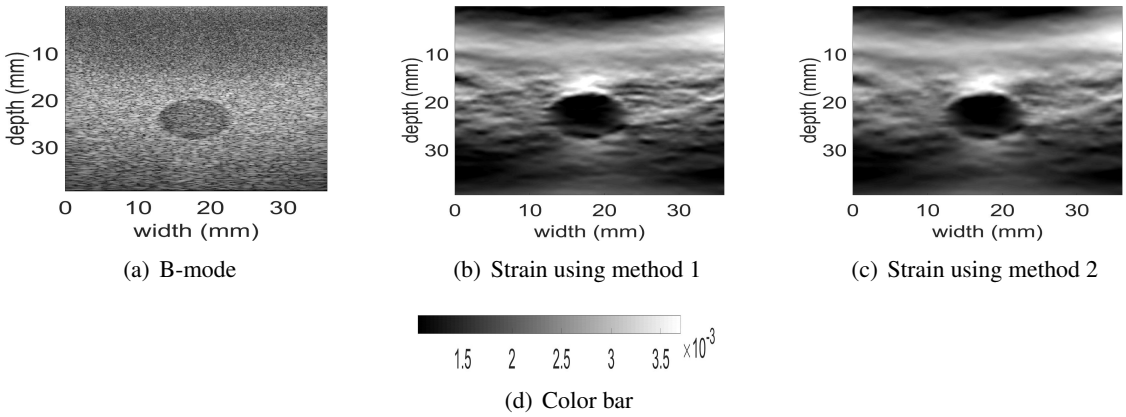
(b) Strain using $p = 2$

(c) Strain using $p = 5$

(d) Strain using $p = 10$

(e) Color bar

Figure A.6: The B-mode ultrasound and axial strain image using PCA-GLUE for *in vivo* liver data after ablation as we increase the number of RF lines p from 2 to 10. The color bar is for the strain images only.



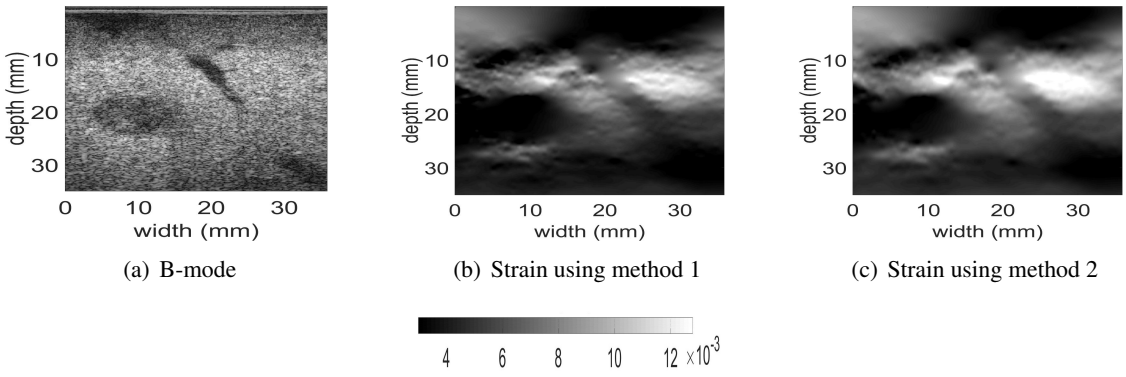
(a) B-mode

(b) Strain using method 1

(c) Strain using method 2

(d) Color bar

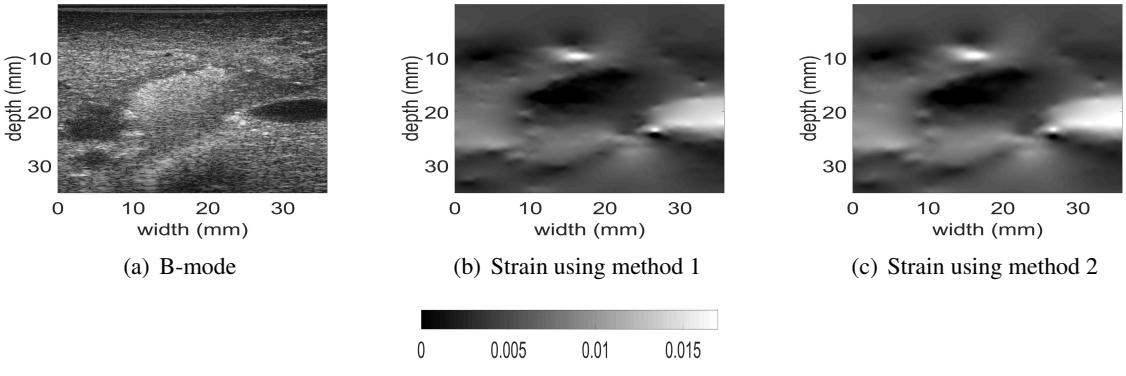
Figure A.7: A comparison between the axial strain estimated using 2 methods for the phantom experiment. In method 1, the lateral displacement given to GLUE is obtained by passing on all RF lines. In the method 2, the lateral displacement given to GLUE is obtained by passing only on 5 RF lines, followed by bi-linear interpolation. The color bar is for the strain images only.



(a) B-mode (b) Strain using method 1 (c) Strain using method 2

(d) Color bar

Figure A.8: A comparison between the axial strain estimated using 2 methods for the *in vivo* liver data before ablation. In method 1, the lateral displacement given to GLUE is obtained by passing on all RF lines. In method 2, the lateral displacement given to GLUE is obtained by passing only on 5 RF lines, followed by bi-linear interpolation. The color bar is for the strain images only.



(a) B-mode (b) Strain using method 1 (c) Strain using method 2

(d) Color bar

Figure A.9: A comparison between the axial strain estimated using 2 methods for the *in vivo* liver data after ablation. In method 1, the lateral displacement given to GLUE is obtained by passing on all RF lines. In method 2, the lateral displacement given to GLUE is obtained by passing only on 5 RF lines, followed by bi-linear interpolation. The color bar is for the strain images only.

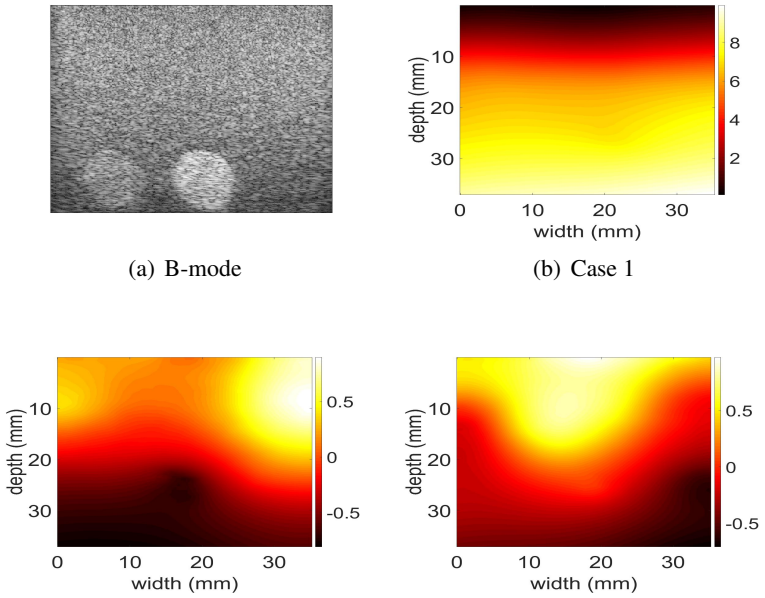
Appendix B

Supplementary Material for the MLP Classifier

In this appendix, we perform an experiment where we give out-of-plane RF frame pairs (i.e. poor) to the multi-layer perceptron (MLP) classifier, to see whether it can correctly classify them as bad pairs.

B.1 Results

We collected new data where the probe displacement is completely out-of-plane, trying to mimic an inexperienced user. We found that out of a total of 160 out-of-plane RF frame pairs, our classifier only chose 11 pairs as suitable for elastography, achieving an accuracy of 93.125%. This shows the ability of our classifier to distinguish the RF frames collected by an inexperienced user. Fig. B.1 shows 3 of the 11 cases that were classified as suitable RF pairs. Although the compression was not in the axial direction, the displacement images are not completely out-of-plane.



(a) B-mode (b) Case 1

(c) Case 2 (d) Case 3

Figure B.1: Some of the displacement images incorrectly classified as suitable for elastography.