Guided Ultrasound Imaging using a Deep Regression Network

Jenish Marharjan[1], Benjamin R. Mitchell[1], Vincent W. S. Chan[2], Edward Kim[3]

[1]Department of Computing Sciences, Villanova University, Villanova, PA
[2]Department of Anesthesia, University of Toronto, ON
[3]Department of Computer Science, Drexel University, Philadelphia, PA

ABSTRACT

In this work, we present a machine learning method to guide an ultrasound operator towards a selected area of interest. Unlike other automatic medical imaging methods, ultrasound imaging is one of the few imaging modalities where the operator's skill and training are critical in obtaining high quality images. Additionally, due to recent advances in affordability and portability of ultrasound technology, its utilization by non-experts has increased. Thus, there is a growing need for intelligent systems that have the ability to assist ultrasound operators in both clinical and non-clinical scenarios. We propose a system that leverages machine learning to map real time ultrasound scans to transformation vectors that can guide a user to a target organ or anatomical structure. We present a unique training system that passively collects supervised training data from an expert sonographer and uses this data to train a deep regression network. Our results show that we are able to recognize anatomical structure through the use of ultrasound imaging and give the user guidance toward obtaining an ideal image.

Keywords: deep learning, guided ultrasound imaging, transfer learning, regression

1. INTRODUCTION

Ultrasound imaging is a safe and efficient imaging modality that has been in use for clinical diagnosis for decades.[1–3] Ultrasound is unlike other automated imaging modalities (MRI, Xray, PET) because the quality of ultrasound images is highly operator-dependent. Therefore a well trained sonographer is critical to fully realizing the diagnostic capabilities of ultrasonography.[4] Computer Aided Diagnostic (CAD) systems supported by machine learning technology have successfully been implemented for assisting clinical diagnosis in a variety of settings, including breast tumor classification, liver diseases, and thyroid nodule diagnosis.[5] CAD systems have thus become a powerful tool for clinicians in those settings. In our project, we aim to leverage machine learning to assist an ultrasound operator in finding a selected area of interest, such as a particular organs, nerves, arteries, etc..

The successful implementation of this machine learning model would have widespread impact in both clinical and non-clinical settings. With recent advances in the ultrasound technology, ultrasound devices have become more affordable, durable and portable resulting in the growth of its utilization by non-radiologist physicians.[6] Advancements in the ultrasound technology and machine learning support CAD systems can be leveraged to make ultrasound imaging accessible for utilization by non-experts. In order to train such a system, we developed a framework that is able to collect supervised training data in a passive manner, alleviating the costly burden on physicians for labeling thousands of ultrasound images and video. We also leverage machine learning techniques such as transfer learning to utilize neural network weights that have been pre-trained on millions of images in order to boost the accuracy of our system.

2. BACKGROUND

With recent advancements in deep learning and computer vision, especially convolutional neural networks (CNNs), there has been a sharp rise in the use of CNNs for medical imaging in various tasks, i.e. image classification, detection, enhancement, image generation, registration and segmentation.[7] CNNs are a type of deep learning model used for identifying patterns in grid-like data such as 2D representation of images.[8] A CNN consists mainly of three types of layers: convolution and pooling layers that help extract the features from the image and a fully connected dense layer that maps the extracted features to the final output. The ability of a

CNN to extract and identify features from anywhere in the image makes it highly efficient in image processing.[8] In the past few years several CNN architectures have been published that show impressive performance on a variety of image based tasks. Some of the notable CNN architectures include but are not limited to VGG-16,[9] Inception-v3,[10] and ResNet-50.[11]

2.1 Network Architecture

For our work, we selected the VGG-16 CNN architecture. VGGNet is among the influential networks developed by Simonyan et al.[9] that presented an effective design principle for CNN architectures. ZfNet previously had suggested that the performance of CNNs could be improved by using smaller filter sizes[12] which VGG experimentally demonstrated by replacing filters of higher order by 3x3 filters. Small size filters reduced the number of free parameters and consequently the computational complexity of the optimization problem, starting a trend in CNN research to work with smaller filters. VGG was able to perform well for image classification as well as localization problems.[13] Three models, varying only in the number of convolutional layers, were proposed - VGG-11, VGG-16, and VGG-19 with 11, 16, and 19 layers respectively.[9]

2.2 Transfer Learning

Deep learning methods need massive amounts of data to discover patterns in the data, which makes them highly data dependent.[14] Since collection of data is expensive and complex, some domains face the inevitable problem of insufficient training data.[15] One solution to this issue is transfer learning. Transfer learning solves the problem of insufficient training data by using knowledge learnt in one domain (source) to another (target). Transfer learning relaxes the assumption that the training data and the test data must be from the *same* domain, replacing it with the weaker assumption that the domains are merely similar.[16] One of the approaches for deep transfer learning is network-based deep transfer learning which is, "similar to the processing mechanism of the human brain, and it is an iterative and continuous abstraction process. The front-layers of the network can be treated as a feature extractor, and the extracted features are versatile".[15]

One of the many domains that faces the problem of insufficient training data is medical imaging. Collecting and labeling biomedical images is a tedious, time consuming and costly task that demands domain specific knowledge and skills which are difficult to acquire.[17] Many researchers have demonstrated the use of transfer learning by using fine-tuned CNNs for biomedical image analysis.[18–20] In our approach, we use transfer learning by freezing the pre-trained convolutional layers of CNNs trained on ImageNet and only training two new fully connected layers at the top.

2.3 Deep Regression Models

Since the goal for our framework is to predict continuous values, we employ a regression technique. Traditionally, deep learning architectures have shown high performance in computer vision tasks such as image classification or object detection but there are many scenarios such as head-pose estimation,[21] facial landmark detection,[22] human pose estimation,[23] or age estimation[24] where the deep learning architectures are used to solve regression problems. In most of these scenarios, the softmax layer of the deep learning models used for classification are replaced by a fully connected regression layer with linear or sigmoid activation,[25] and this is the approach we follow.

3. DESCRIPTION OF PURPOSE

Our work has two primary goals. The first goal is to create a dataset of ultrasound images with their corresponding transformation information. We use an external device to record six degrees of freedom from the hand of the sonographer, e.g. x position, y position, z position, yaw, pitch, and roll orientation. The transformation data is then associated with an ultrasound image captured by the probe. For our prototype we used two off-the-shelf devices to detect the position of the probe. The first is a visual only device that detects the position and orientation of a persons hand called the Leap Motion Device*. The second device is an active tracking device

---

*Leapmotion. https://leapmotion.com/
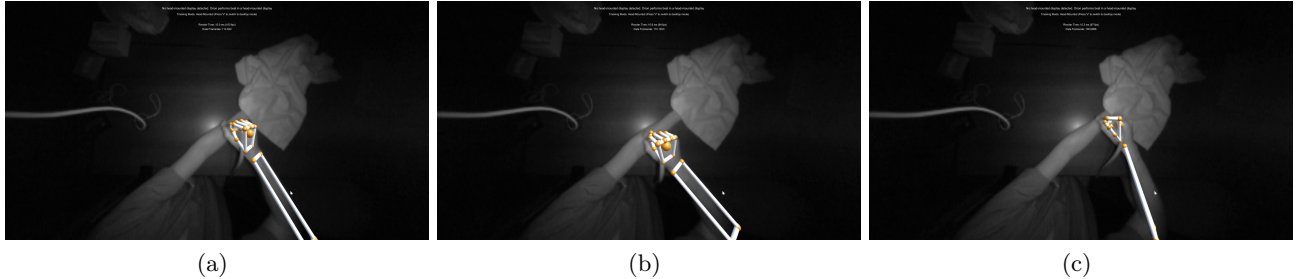
| (a) | (b) | (c) |

Figure 1. Visualization of tracking the hand skeleton using a Leap motion device. The Leap device is mounted above, facing down towards the scanning plane. The physician is scanning for the median nerve using an ultrasound probe and the position of the physician's hand is being tracked and recorded through the Leap motion SDK. The positional vectors are timestamp matched to ultrasound images for training of the deep regression network.

with base station called the Vive Tracker[†]. Both devices are low cost methods of gathering the transform of the sonographer's hand during the ultrasound imaging process.

The second goal is to train a deep neural network that can map an input ultrasound image to a set of values representing a vector displacement from the ideal area of interest. This displacement can be used to guide the operator's hand toward the desired location. Mathematically, this is a regression problem where inputs will be the ultrasound images and the outputs will the set of real-valued vector coordinates. For the neural network, we utilize transfer learning, widely used in computer vision, by fine-tuning a model pre-trained on a source task using training images from the target task.

4. METHOD

For our methodology, we focus on the localization of the median nerve and radial artery in the arm. The median nerve runs from the forearm to into the wrist (carpal tunnel) then into the hand. It provides sensation to the palm and fingers. When the nerve is compressed, it can lead to tingling, numbness, or weakness in the hand. Ultrasound imaging of this region is necessary for proper diagnosis of carpal tunnel syndrome. In other cases, localization by an anesthesiologist may be necessary when performing an median nerve block during surgery or treatment.

4.1 Passive Collection of Supervised Data

For our regression problem, the training dataset requires a set of ultrasound images of and around the median nerve, each labeled with a set of transformation values that "point" towards the ideal image. If we consider the ideal image as the origin, we can associate each surrounding ultrasound image with a displacement vector from the origin.

Each subject has their median nerve scanned for approximately 60 seconds. During this time, the sonographer scanned back and forth around the area of interest. This process captured the surrounding anatomy of the area while positional data of the sonographer's hand was being tracked. Around the 45 second mark, the sonographer located and identified the best image of the nerve and remain fixated on this area as time expired. Thus, the best image was the last image in the collection sequence, and the "origin" transformation vector was the vector associated with the last image frame. We then can use this vector as the relative origin for all other transformation data previously collected during the ultrasound scan.

The ultrasound imaging data is captured using a SonoSite Edge II ultrasound machine. This machine can store video at a resolution of 640x480. The video captured comes in at 7.5 frames per second. The probe used for the experiments is a Linear probe (6-13 MHz). For the purposes of this study, we remove all personal information from the video (e.g. patient name, date, and other associated metadata), and crop out just the ultrasound image portion. This center region is used as the input image to the neural network and has a resolution of 525x255. For

---

[†]Vive Tracker. https://www.vive.com/us/vive-tracker/

(a) HTC Vive tracker

(b) HTC Vive base station

Figure 2. Images of the experimental process using the HTC Vive tracker and base station. In (a), we capture both the ultrasound image and six degrees of freedom from the Vive tracker strapped to the sonographer's hand. In (b), we show the Vive base station required as a frame of reference for the tracker. The base station must be within line of sight of the tracker unit.

the transformation data, we experimented with two devices. One was the Leap motion device and the second was the Vive Tracker.

A Leap Motion Controller is a motion sensor that uses an infrared scanner and sensor to map and track the human hand. It uses video only, and does not require any additional hardware. A Leap Motion Controller's raw sensor capability has been validated as reliable.[26] It has also been determined that a Leap Motion Controller can relay static positional data with a standard deviation of less than 0.5mm.[27] We use the Leap Motion Controller to track and capture the position of the sonographer's hand holding the ultrasound probe. See Figure 1 for a visualization of the Leap motion capture system.

The Vive tracker is a motion accessory that was designed to track physical controllers in the real world while the user is immersed in virtual reality. However, we repurposed this technology to track without a VR headset using the pyopenvr python library.

The Vive uses a "lighthouse" tracking system that consists of base stations which emit timed infrared pulses at 60 pulses per second. These are then received by the tracker to produce measurements with sub-millimeter precision. Valve's lighthouse boxes use spinning LEDs and active laser emitters that flash a beam of light across the tracking space. The tracker has photosensors that detect the flashes and the laser beams. Using the laser time of travel, the photosensors can compute their exact position relative to the base stations in the room. An example of the Vive setup can be seen in Figure 2.

The images captured from the ultrasound probe and the positional information obtained from the Leap Motion Controller or Vive tracker are mapped based on their corresponding timestamps to get the final dataset. This system is able to passively collect supervised training data by simply watching/recording the sonographer scan for a specific area of interest and specifying which image is the ideal image.

4.2 Training the Deep Regression Model

For our machine learning framework, we implemented a VGG16 model using the Keras‡ library, fine-tuned to ultrasound images. VGG16 is a convolutional neural network with a stack of 16 convolutional layers followed by three fully connected layers and a final soft-max layer.[9] The soft-max layer makes it fit for classification problem but not for regression. Thus, in order to fine tune the model, we froze all the pre-existing layers and removed the soft-max output layer.

Next, we created 2 models, one for the positional regression, and one for the orientation regression task. The models were modified to contain a fully connected layer of 128 neurons with a tanh activation function, a

---

‡Keras Deep Learning Framework https://keras.io/

dropout of 0.2, and then a final output of size 3 to correspond to the real valued vector parameters. Ultimately, our model was able to utilized transfer learning, allowing us to use a deep learning model pre-trained with a huge dataset of real world images (ImageNet) on our task of regression on ultrasound images.

5. EXPERIMENTS AND RESULTS

5.1 Data Collection Comparison

In this section we present qualitative observations on the data collection devices that we tested, the Leap Motion and the Vive Tracker. Table 1 delineates the positive and negative experiences we encountered during testing.

Feature	Leap Motion (vision only)	Vive Tracker (hardware)
Cost	The leap motion is a very low cost ($<$\$100 USD) and fairly accurate device	The vive tracker is relatively low cost but requires the purchase of trackers as well as a base station ($<$\$350 USD).
Tracking	Tracking is accurate with the leap motion when the device has a clear, unobstructed view of the sonographer's hand. Occasionally, the leap motion would lose sight of the hand during the experimental procedure.	The vive tracker is strapped onto the sonographer's hand. If the line of sight between the base station and the tracker is completely obstructed, this could result in a loss of tracking; however, in our experiments, this never occurred.
Ease of Use	Setup requires mounting the leap motion above the examination plane. Sonographer does not need to wear any tracking device; however, must pay careful attention that the leap can see and track their hand.	Base station can be placed anywhere with line of sight. Sonographer must strap the tracking device to their hand.
Overall	Overall, the leap motion provided an inadequate data collection experience. The robustness of the method was not useabled for our final framework.	The vive tracker was robust and fairly easy to use. The tracker strapped to the sonographer's hand did not hinder the scanning process. Although a little more expensive, this device was preferred.

Table 1. Feature comparison between the Leap motion device and the HTC Vive tracker. Ultimately, the Vive tracker provided a more robust tracking experience.

After these preliminary observations, we decided to move forward with data collection using the HTC Vive Tracker. When experimenting with the Leap Motion, we had the occasional loss of tracking occur. This would happen when the physician's hand over rotated and obscured the physician fingers. Without these landmarks, the vision only system could not recover the correct pose.

Additionally, and quite unexpectedly, the values being tracked with the Leap motion and vision only system would jump significantly. Upon further inspection, we noticed that the hand being tracked would jump from the physician hand to the imaging subject's hand. In order to solve this, we had to obscure the subject's hand during the experiment with a towel as seen in Figure 1. While this resolved this problem, we note that the vision only system was not robust enough in general for our research.

5.2 Experimental Setup

For our experiments, we collected a total of 3,722 ultrasound images with transformational data around the median nerve in 10 different subjects (subject 8 was ultimately left out of training due to timestamp discrepancies). We used a leave-one-out method in order to train, test, and evaluate our model. With the leave-one-out subject model, the entire set of the testing subject's images were not seen by the model at training time. A sample 60 second capture is shown in Figure 3.
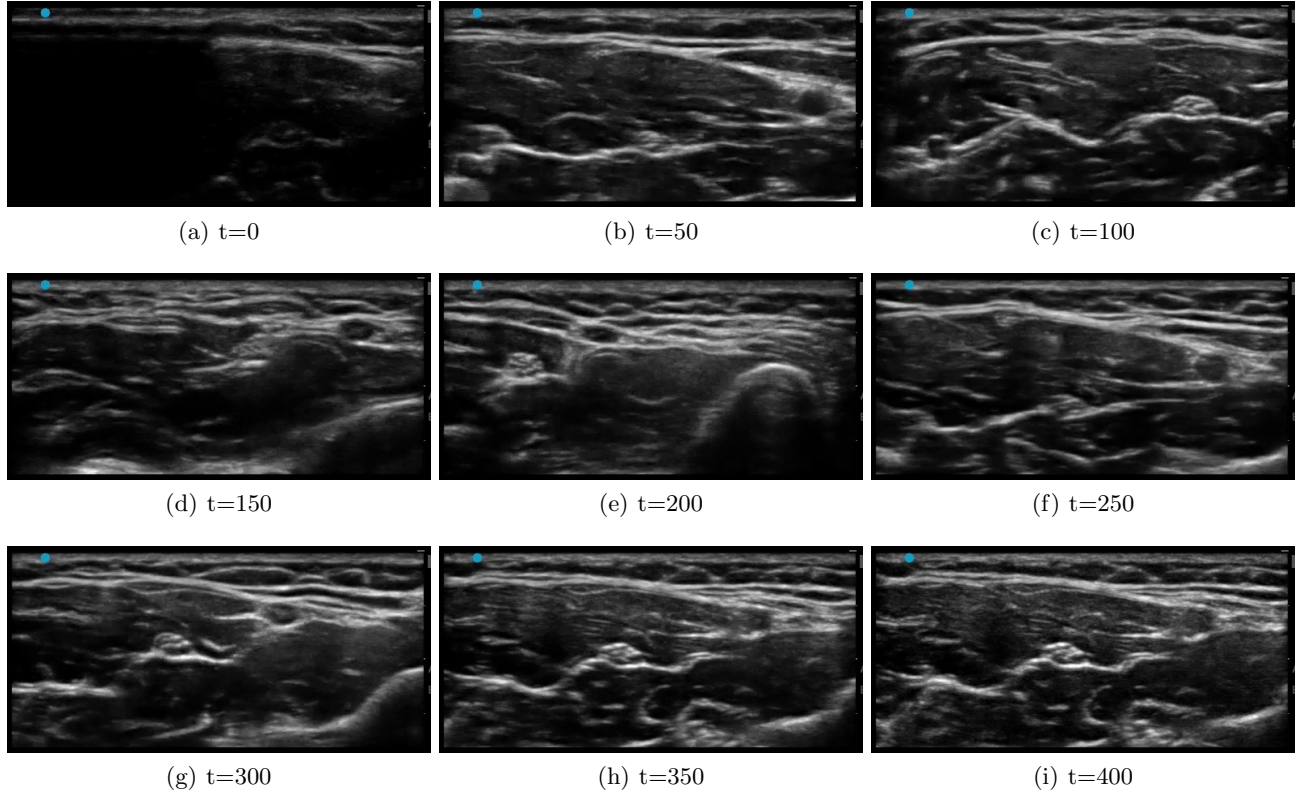
Figure 3. Images captured during a scan for the median nerve and radial artery. Images (a)-(g) are images that have been captured at a frame times 0-300, at intervals of 50. These images are used for training the regression network. Each of these images are associated with a transformation vector that measures its displacement from the ideal result. Image (h)&(i) show the ideal result with the median nerve centered and the radial artery clearly visible in the image.
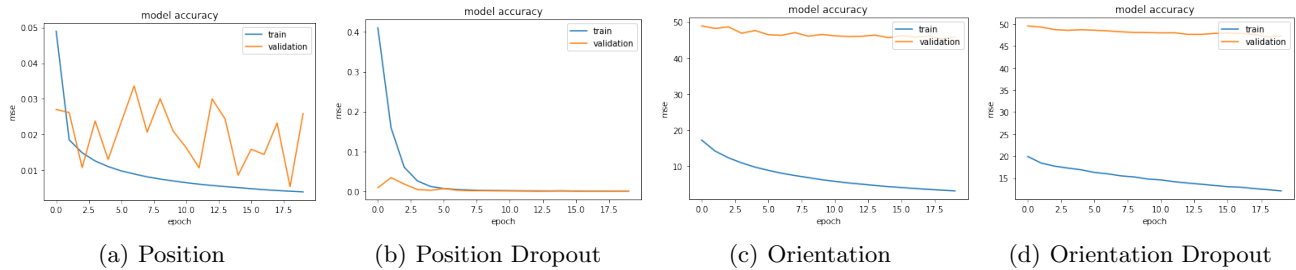

Figure 4. Training and validation results with s-fold cross validation, and when s=10. The graphs (a) & (b) show the translation loss smooths out when using dropout in training. A similar effect can be seen with the orientation in (c) & (d). Note that dropout is always turned off during validation and testing.

As mentioned above, we used the HTC Vive Tracker for data collection. The tracker is oriented such that the X axis is parallel to the forearm, the Z axis is perpendicular to the forearm, and the Y axis is vertical to the arm. We define the Pitch as the angle that rotates around the X axis, the Yaw as the angle that rotates around the Y axis, and the Roll as the angle that rotates around the Z axis.

5.3 Model Performance

To train the neural network models, we performed a small amount of data augmentation on the input - rotation range of 2 degrees, width, height, and zoom shifts of up to 2 percent. Interestingly, more data augmentation did not help the system generalize to unseen data (in fact it prevented the system from reducing the training loss).

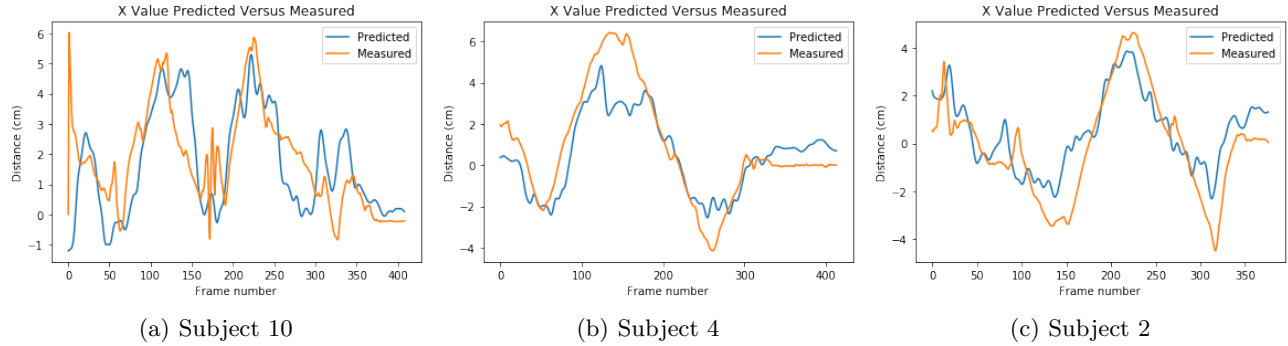(a) Subject 10   (b) Subject 4   (c) Subject 2

Figure 5. Graphs illustrating the difference between the actual measured distance to the target area of interest and the predicted distance by the trained model for the X axis (parallel to the median nerve) on different subjects. The video spans approximately 60 seconds of scanning for each subject and the distance is reported in centimeters.

We also chose to create two models, one for translation, and one for orientation due to the scale discrepancy of outputs (meters vs degrees). The magnitude of degrees was much larger than the translation and thus dominated the gradient in a model where both features were predicted simultaneously. We train the model for 40 epochs using a mean squared error loss. The training time on an Intel I9 3.6 Ghz CPU with an Nvidia RTX 2080 is approximately 14.6 minutes per fold. Sample training and validation error can be seen in Figure 4 where we highlight the smoothness of validation when using dropout in the model.

Quantitatively, our model was able to predict the transformation vectors for each degree of freedom. Here, although we did train and predict for the Y axis, we realized that this direction does not support the ultrasound scanning procedure. We also note that the Y position did not correspond to pressure, and should be disregarded by the ultrasound operator. Thus, we do not report the Y regression degree of freedom as the probe should always remain in contact with the subject's arm. The mean and standard deviation on this dataset of each of the degrees of freedom is the following:

- **Mean X distance error is 1.19 cm ± 0.98 cm** (parallel to median nerve)
- **Mean Z distance error is 1.78 cm ± 1.32 cm** (perpendicular to median nerve)
- **Mean Yaw distance error is 4.38 degrees ± 3.20 degrees** (axis around Y)
- **Mean Pitch distance error is 4.26 degrees ± 3.58 degrees** (axis around X)
- **Mean Roll distance error is 4.22 degrees ± 2.67 degrees** (axis around Z)

In addition, we present graphs of the predicted and ground truth values as a function of time in Figure 5. The predicted line show is smoothed by a hamming convolutional window over 15 frames. This signal convolution was necessary to provide some temporal consistency to the neural network output. As demonstrated, the model is generally able to accurately predict the displacement vector from the ideal ultrasound image.

5.4 Limitations

For this study, we had a small number of samples (N=10) and there is a significant amount of variation between subjects. In future work, we anticipate that with a greater number of subjects the training distribution would better represent the testing distribution and the model should generalize better. Also, finding an adequate image of the median nerve is an easier task than many other internal organs such as the liver and heart. We are encouraged by our current research and proof of concept, but would need a more extensive evaluation to extend this work to other organs. Finally, there are more degrees of freedom involved in the scanning process. One particularly important factor is the pressure that the sonographer applies to the target area. This is a difficult measurement to capture without actually modifying the ultrasound probe, and is something we would like to address in future work. Being able to capture more data from different target areas of the body with high precision would be the key to translating this research to the clinical setting.

6. CONCLUSION

In conclusion, we present a guided ultrasound imaging framework using a deep regression neural network. The training data is obtained passively, by simply observing the ultrasound sonographer during the scanning process. We tested several tracking methods and presented the benefits of each of method. Next, we captured transformation data using the HTC Vive Tracker and matched the data to ultrasound images via their corresponding timestamps. We then trained a deep neural network using transfer learning and fine-tuning techniques to regress to the optimal transformation vector. Our quantitative and qualitative results show that we are able to accurately estimate these vectors when finding the median nerve in a subject's arm.

REFERENCES

[1] Drukker, K., Giger, M. L., Horsch, K., Kupinski, M. A., Vyborny, C. J., and Mendelson, E. B., "Computerized lesion detection on breast ultrasound," *Medical physics* **29**(7), 1438–1446 (2002).

[2] Sahiner, B., Chan, H.-P., Roubidoux, M. A., Hadjiiski, L. M., Helvie, M. A., Paramagul, C., Bailey, J., Nees, A. V., and Blane, C., "Malignant and benign breast masses on 3d us volumetric images: effect of computer-aided diagnosis on radiologist accuracy," *Radiology* **242**(3), 716–724 (2007).

[3] Chen, C.-M., Chou, Y.-H., Han, K.-C., Hung, G.-S., Tiu, C.-M., Chiou, H.-J., and Chiou, S.-Y., "Breast lesions on sonograms: computer-aided diagnosis with nearly setting-independent features and artificial neural networks," *Radiology* **226**(2), 504–514 (2003).

[4] Pinto, A., Pinto, F., Faggian, A., Rubini, G., Caranci, F., Macarini, L., Genovese, E. A., and Brunese, L., "Sources of error in emergency ultrasonography," *Critical ultrasound journal* **5**(S1), S1 (2013).

[5] Kim, E., Baloch, Z., and Kim, C. S., "Computer assisted detection and analysis of tall cell variant papillary thyroid carcinoma in histological images," in [*SPIE Medical Imaging*], 94200A–94200A, International Society for Optics and Photonics (2015).

[6] Shah, S. P., Epino, H., Bukhman, G., Umulisa, I., Dushimiyimana, J., Reichman, A., and Noble, V. E., "Impact of the introduction of ultrasound services in a limited resource setting: rural rwanda 2008," *BMC international health and human rights* **9**(1), 4 (2009).

[7] Sengupta, S., Basak, S., Saikia, P., Paul, S., Tsalavoutis, V., Atiah, F., Ravi, V., and Peters, A., "A review of deep learning with special emphasis on architectures, applications and recent trends," (2019).

[8] Yamashita, R., Nishio, M., Do, R., and Togashi, K., "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging* **9** (06 2018).

[9] Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556* (2014).

[10] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A., "Going deeper with convolutions," (2014).

[11] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," (2015).

[12] Zeiler, M. D. and Fergus, R., "Visualizing and understanding convolutional networks," (2013).

[13] Khan, A., Sohail, A., Zahoora, U., and Qureshi, A. S., "A survey of the recent architectures of deep convolutional neural networks," (2019).

[14] Kim, E., Mente, S. L. D., Keenan, A., and Gehlot, V., "Digital pathology annotation data for improved deep neural network classification," in [*Medical Imaging 2017: Imaging Informatics for Healthcare, Research, and Applications*], **10138**, 101380D, International Society for Optics and Photonics (2017).

[15] Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C., "A survey on deep transfer learning," in [*International Conference on Artificial Neural Networks*], 270–279, Springer (2018).

[16] Weiss, K., Khoshgoftaar, T.M., Wang, and D., "A survey of transfer learning.," (2016).

[17] Zhou, Z., Shin, J., Zhang, L., Gurudu, S., Gotway, M., and Liang, J., "Fine-tuning convolutional neural networks for biomedical image analysis: Actively and incrementally," *Proceedings. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2017** (07 2017).

[18] Carneiro, G., Nascimento, J., and Bradley, A. P., "Unregistered multiview mammogram analysis with pretrained deep learning models," in [*Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*], Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., eds., 652–660, Springer International Publishing, Cham (2015).

[19] Chen, H., Dou, Q., Ni, D., Cheng, J.-Z., Qin, J., Li, S., and Heng, P.-A., "Automatic fetal ultrasound standard plane detection using knowledge transferred recurrent neural networks," **9349**, 507–514 (10 2015).

[20] Gao, M., Bagci, U., Lu, L., Wu, A., Buty, M., Shin, H.-C., Roth, H., Papadakis, G., Depeursinge, A., Summers, R., Xu, Z., and Mollura, D., "Holistic classification of ct attenuation patterns for interstitial lung diseases via deep convolutional neural networks," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging Visualization* **6**, 1–6 (06 2016).

[21] Burgos-Artizzu, X., Perona, P., and Dollár, P., "Robust face landmark estimation under occlusion," *Proceedings of the IEEE International Conference on Computer Vision* , 1513–1520 (12 2013).

[22] Fanelli, G., Gall, J., and Van Gool, L., "Real time head pose estimation with random regression forests," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **617**, 617–624 (06 2011).

[23] Zhu, X. and Ramanan, D., "Face detection, pose estimation, and landmark localization in the wild," *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* , 2879–2886 (06 2012).

[24] Guo, G., Fu, Y., Dyer, C., and Huang, T., "Image-based human age estimation by manifold learning and locally adjusted robust regression," *Image Processing, IEEE Transactions on* **17**, 1178 – 1188 (08 2008).

[25] Lathuilière, S., Mesejo, P., Alameda-Pineda, X., and Horaud, R., "A comprehensive analysis of deep regression," (2018).

[26] Smeragliuolo, A. H., Hill, N. J., Disla, L., and Putrino, D., "Validation of the leap motion controller using markered motion capture technology," *Journal of biomechanics* **49**(9), 1742–1750 (2016).

[27] Guna, J., Jakus, G., Pogačnik, M., Tomažič, S., and Sodnik, J., "An analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking," *Sensors* **14**(2), 3702–3720 (2014).