OBJECT DETECTION AND TRACKING IN ULTRASOUND SCANS USING AN OPTICAL FLOW AND SEMANTIC SEGMENTATION FRAMEWORK BASED ON CONVOLUTIONAL NEURAL NETWORKS

Abdullah F. Al-Battal[1,2], *Imanuel R. Lerman*[1,3], *Truong Q. Nguyen*[1]

[1] Electrical and Computer Engineering Department, UC San Diego, La Jolla, California, USA
[2] Electrical Engineering Department, King Fahd University of Petroleumand Minerals, Saudi Arabia
[3] School of Medicine, UC San Diego, La Jolla, California, USA

ABSTRACT

Based on non-ionizing radiation, ultrasound scanning is safe to image a specific region of the body repeatedly to identify and localize target anatomical structures during therapeutic and diagnostic procedures. However, it is labor intensive, and requires sonographers to have extensive experience to be able to identify and track these anatomical structures of interest, making the identification and tracking process highly prone to errors. In this paper, we propose a framework to autonomously detect, localize and track anatomical structures in ultrasound scans during scanning and therapeutic sessions in real-time. The proposed framework uses a segmentation-based convolutional neural network (CNN) to detect and localize the target anatomical structure within a scan. Concurrently, it uses an optical flow CNN to track the movement of this structure across frames to accurately guide therapeutic procedures. We tested the framework on detecting and tracking the Vagus nerve in ultrasound scans. It achieved state-of-the art localization and tracking accuracy with an average error of less than 1.25 mm for localization and 0.75 mm for tracking while maintaining an inference time of less than 35 ms.

***Index Terms*—** Real-time object detection, real-time object tracking, ultrasound, Vagus nerve, convolutional neural networks

1. INTRODUCTION

Ultrasound scanning is used in many medical settings to image and identify target anatomical structures, especially in soft tissue as it provides the required contrast to distinguish different tissues for diagnostic and therapeutic purposes. Based on non-ionizing radiation, ultrasound scanning is safe to image a specific region of the body repeatedly to identify and localize target anatomical structures [1, 2]. While scanning, a typical ultrasound device generates 20 to 30 scans per second. Throughout a scanning session, a sonographer uses these scans to identify the target anatomical structures, which is labor intensive as each of these scanning sessions can take up to 60 minutes. Sonographers also need to have extensive experience to identify the underlying anatomical structure of interest accurately and efficiently, which is why novice sonographers make 52% more diagnostic and anatomical structures identification errors than expert sonographers [3]. Internal and external body movements, such as breathing, cause tissue and organ movements. While sonographer transducer to skin pressure and inadvertent transducer sliding inevitably change the scan window. These movements are a major challenge in therapeutic procedures including radiation therapy and neurological simulations where accurate localization of target anatomical structures is key to their success [4]. Therefore, an autonomous framework that can detect, localize, and track anatomical structures of interest would significantly assist sonographers in performing these procedures.

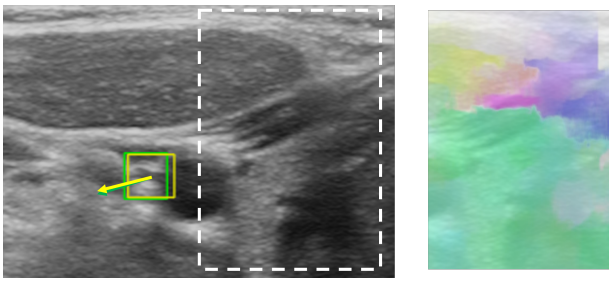

Fig. 1: An ultrasound scan of the Vagus nerve. Left: Ground truth (green) and predicted (yellow) bounding boxes with movement estimation arrows using optical flow (arrows length amplified for display). Right: Cutout bounded by dashed box with the optical flow map represented by RGB colors overlaid.

For tracking, methods such as region of interest (ROI) matching have been designed to track ROIs across frames using block matching [5, 6]. Shape fitting has also been used to track objects across images, such as fitting to circles and ellipses to track veins and arteries [7]. Some designs used optical flow algorithms to estimate movement [8, 9] and track objects, while others used supporters [10]. More recently deep learning approaches have been designed to solve this tracking problem using Siamese networks [11, 12]. These methods perform well under established assumptions, but they currently do not interface with the sonographer to allow for real-time object detection and tracking, due to: 1) inherent slow tracking speeds [11, 13], or 2) a need for operator dependent identification of the target ROI [5], or 3) in some cases both.

For detection and localization, segmentation-based approaches have been historically used in medical imaging to identify a pixel-wise map of the parts in an image that correspond to a target object [14]. Prior to the era of deep learning, these segmentation approaches were very limited in scope and application where they failed to generalize beyond the data they were designed for [15]. Deep learning approaches were able to improve the generalizability and accuracy of segmentation approaches at the expense of computational complexity and memory usage [16]. Nevertheless, these approaches require pixel-wise annotations to create the training dataset, which is a labor-intensive operation and is hard to scale.

This work was supported in part by the Biological Advanced Research and Development Authority (BARDA Contract: 7515011900038), National Institute of Health (NIH Contract: IK2RX002920) and the David and Janice Katz Neural Sensor Research Fund in Memory of Allen E. Wolf.

In this paper, we propose a framework that is capable of real-time object detection and localization using a weakly trained segmentation-based CNN accompanied by an optical flow CNN that can track the target object location across frames by estimating its movements. The framework can detect and localize a target anatomical structure within the ultrasound scan field of view with an average localization error of less than 1.25 mm, and tracking error of less than 0.75 mm. The framework can operate autonomously by first detecting the target object, then tracking it across frames, unlike other previously proposed frameworks. To track an object, other frameworks depended on a sonographer to manually identify it first, hindering their ability to assist sonographers in reducing detection and localization errors. [11, 12, 13]. We test our framework on detecting and tracking the cervical Vagus nerve encased within Carotid artery sheath, which is located at a variable and dynamic depth of approximately 1.2 to 2.5 cm (depending on transducer probe to skin pressure and compressions in surrounding arteries and veins) [17, 18, 19]. We demonstrate that the proposed framework outperforms real-time state-of-the art object detection models (YOLOv4 [20] and EffecientDet [21]) in detecting and localizing the Vagus nerve. It also demonstrates a state-of-the-art object tracking accuracy in ultrasound images when compared to current tracking methods [12].

2. RELATED WORKS

Current best performing object detection models employ convolutional neural networks to extract features, then localize these features to estimate the bounding box that encapsulates the desired object [20, 22]. They are trained to estimate the 4 coordinates of a bounding box and require a large training set as they tend to overfit when trained on smaller training sets [23]. In medical imaging, datasets are usually small in size (in the range of hundreds to few thousand images) [24]. Therefore, in medical imaging, semantic segmentation is used more frequently, where a pixel-wise segmentation map is generated, and models can be trained on smaller datasets without overfitting [14]. However, segmentation requires training sets with detailed pixel-wise annotations that are labor intensive and expensive to generate. In [25], we proposed a segmentation-based framework that uses a modified U-Net [14] for object detection. It was trained on pixel-wise segmentation maps generated from bounding box information, negating the need for expensive-to-generate detailed pixel-wise annotations.

Tracking objects of interest across frames in ultrasound images has been historically approached through region similarity matching and shape fitting. Both approaches used extensive search block matching to look for the object of interest within the field of view of a scan [5, 6]. Block matching techniques are inherently slow and cannot operate in real-time, i.e., at 30 frames per second (fps), even with optimized search algorithms [13]. They also tend to fail in tracking non-simple objects, i.e., objects other than arteries, veins and speckles, especially when shape fitting is used for matching. Recently, Siamese neural networks have been deployed to track objects of interest [11, 12]. These networks use a template image of the target object and the ultrasound frame as inputs to estimate a similarity map based on the most probable location of the object [26]. One major shortcoming of these networks is that the template image of the target object must be provided by the sonographer and cannot be detected autonomously by the network [11, 12]. Moreover, a sonographer must identify the template image of the target object at scan start that may result in perpetuation of human error.

3. METHOD

To autonomously detect, localize, and track a target object of interest in ultrasound scans, our framework uses 5 different stages. Fig. 2 shows an overview of the five different stages, how they are connected, and what they are used for.

3.1. Stage 1: Pre-Processing

The first stage of the proposed framework is used to prepare the scans for the detection, optical flow estimation, and classification stages. The current scan together with the previous one is fed to the backbone detection network in stage 2 as well the optical flow estimation in stage 3. Using the current and previous scans as inputs to the detection network improves the accuracy of the network [25]. To minimize the drawbacks of training on a smaller dataset and to prevent overfitting, several methods of data augmentation were used. We implemented randomized geometrical, brightness and contrast transformations to simulate different imaging conditions. We also implemented elastic transformations [27] to simulate the elasticity of soft tissue being imaged. For the backbone detection network, we used mixtures of inputs [25] to better sample the probability distribution of the input-output pairs [28]. Finally, bounding box information of target objects' locations is used to create the masks that will weakly train the backbone detection network. In these masks, pixels within the boundaries of the bounding box are set to 1 indicating the presence of the object.

3.2. Stage 2: Backbone Segmentation-Based Detection Network

In this stage, we designed a modified U-Net architecture. The network has 4 depth levels and a bridge similar to the original U-Net. In each depth level, the convolution block contains 2 convolutional layers. Feature maps at each depth layer has been reduced by a factor of 2 as shown in Fig. 2, which has two major advantages. The network can operate in real-time, and it does not overfit as the number of parameters is reduced making the model suitable for the small medical datasets. Overall, there are 7.8 million parameters in the network compared to 31 million in the original U-Net. On the other hand, this reduction in size usually translates to a reduction in performance, however, we were able to minimize the effect of this reduction in performance by incorporating several techniques into the design of the network [25]. Two-dimensional (2D) dropout layers were used to regularize high correlation among pixels with spatial proximity [29]. Furthermore, a localization promoting loss function based on the Dice coefficient was used to train the network in addition to the binary cross-entropy (BCE) loss function originally used to train U-Net [14]. The overall loss function between the predicted mask ($\widehat{Y}$) and ground truth (Y) is defined as:

$$\mathcal{L}_{obj}(\widehat{Y}, Y) = \alpha_{bce}\mathcal{L}_{bce}(\widehat{Y}, Y) + \alpha_{dice}\mathcal{L}_{D_c}(\widehat{Y}, Y), \quad (1)$$

where $\mathcal{L}_{bce}$ and $\mathcal{L}_{D_c}$ are the BCE loss and Dice loss, respectively, defined in (2) and (4). α_{bce} and α_{dice} are contribution control coefficients. During training, we used $\alpha_{bce} = 0.25$ and $\alpha_{dice} = 1$. The BCE loss calculates the average of the binary cross-entropy between pixels in the predicted and ground truth mask. Consequently, for each pixel of the predicted mask with a value $\hat{y}$ and ground truth value y at location (i, j), where $i = 1, 2, ..., H$, and $j = 1, 2, ..., W$ (H being the height of the mask, and W the width), the BCE loss over a training batch is defined as:

$$\mathcal{L}_{bce}(\widehat{Y}, Y) = -\frac{1}{N}\sum_{n=1}^{N}\left[\frac{1}{M}\sum_{m=1}^{M}\left[\ell\left(\hat{y}_{m,n}, y_{m,n}\right)\right]\right], \quad (2)$$

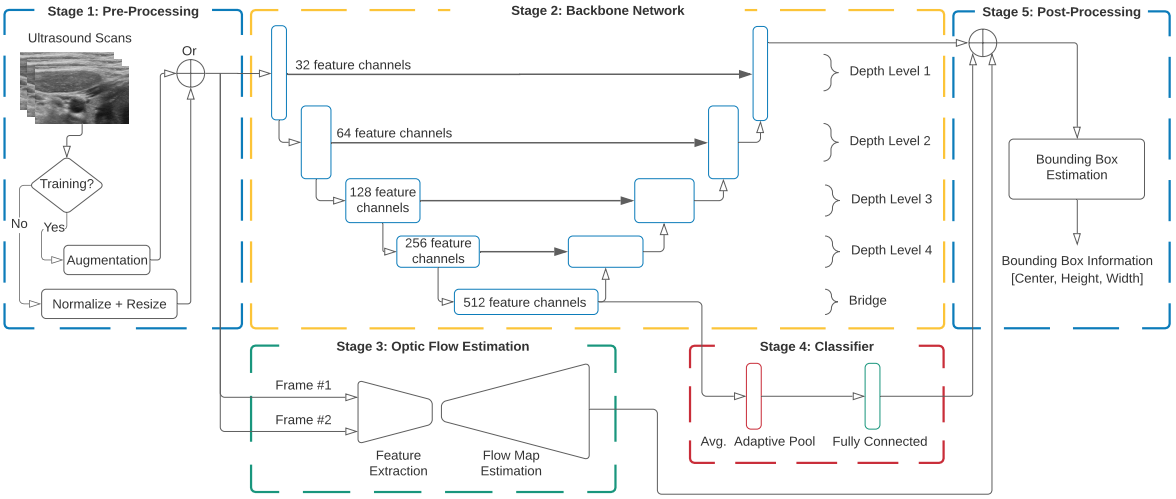

Fig. 2: Overview of the five stages of the proposed object detection and tracking framework.

where N is the number of masks in a batch, M is equal to $H \times W$ (the number of pixels in a mask), $\ell\left(\hat{y}_{m,n}, y_{m,n}\right)$ is the loss computed element-wise between the ground truth and predictions, and is defined as:

$$\ell\left(\hat{y}, y\right) = w_c y \log \sigma(\hat{y}) + (1-y) \log \sigma(\hat{y}), \tag{3}$$

where w_c is a weight that adjusts class c loss contribution to the overall loss, and is used to counter class imbalance in a dataset. $\sigma(\hat{y})$ is the sigmoid function defined as $\sigma(\hat{y}) = 1/(1+exp(-\hat{y}))$. The sigmoid function maps pixels of the mask into a probability space where a $\sigma(\hat{y})$ value larger than a threshold (usually chosen to be 0.5) means that the pixel $\hat{y}$ is part of the target object; otherwise it is considered background. Adjusting w_c and the threshold of $\sigma(\hat{y})$ influence the performance of a network in terms of sensitivity and specificity. The Dice loss can be defined as:

$$\mathcal{L}_{D_c}(\widehat{Y}, Y) = 1 - D_c(\widehat{Y}, Y) \tag{4}$$

This loss penalizes lower Dice coefficient values (D_c), which is defined as [30]:

$$D_c(\widehat{Y}, Y) = \frac{2(\sigma(\widehat{Y}) \odot Y)}{\sum_{m=1}^{M} \sigma(\hat{y}_m) + \sum_{m=1}^{M} y_m}, \tag{5}$$

where $\odot$ represents the element-wise multiplication, and M is equal to $H \times W$ (the number of pixels in a mask).

3.3. Stage 3: Object Tracking Using Optical Flow Estimation

To track the object detected in the 2^{nd} stage as it moves from one frame to another, we incorporated an optical flow estimation network into the design of the framework as the 3^{rd} stage. The network is based on the LiteFlowNet architecture [31]. The architecture has 5.4 million parameters. It takes two frames as inputs and generates an optical flow map representing the movement of each element in frame 1 based on its location in frame 2. The flow map has two components, representing the movement in the x and y directions, respectively [32]. In the feature extraction section of the network, there are six depth levels. In each level, there is a series of convolutional layers followed by leaky ReLU. The number of layers in each level is as follows: 1 layer in the first level, 3 in the second, 2 in each of the third and fourth, and 1 in the fifth and sixth layers. The weights of the feature extraction section are tied for both frames, so, the feature maps at each level will be similar for both frames while differences will be mainly attributed to spatial movements. In the flow map estimation section of the network, feature maps of the second frame ($\mathcal{F}_2$) are warped to the ones of the first frame ($\mathcal{F}_1$) at each level. After warping, displacement-based correlation is calculated between the two feature maps: $\mathcal{F}_1$ and $\tilde{\mathcal{F}}_2$ (the warped $\mathcal{F}_2$). Warping allows the displacements to span a smaller range since the feature maps have been warped, speeding up inference times. This correlation can be defined as:

$$\begin{gathered}\mathcal{F}_c(x, y, k) = \mathcal{F}_1(x, y) \cdot \tilde{\mathcal{F}}_2(x + d_x, y + d_y) \\ \forall\ d_x, d_y \in [-d, d], \quad d \in \mathbb{Z}\end{gathered} \tag{6}$$

In (6), d represents the maximum displacement used to calculate correlation, and k goes from 1 to $(2d+1)^2$, which is the total number of displacements; assuming we are using a stride of 1. The output flow-map generated from stage 3 is represented by a tensor of size $H \times W \times 2$, where at any location (x, y) in the flow map, 2 numbers represent the movement in the x and y directions.

3.4. Stage 4: Frame Classification

Stages 2 and 3 are designed to detect, localize and track an object in scans. Therefore, to identify whether a scan has that object, we use a classifier fine-tuned for this task as shown in Fig. 2. The classifier uses two additional layers. It is fed from the output of the last layer of the bridge in stage 2, which contains 512 feature map channels. These are flattened to a tensor of length 512 using the average global pooling layer that was proposed as part of ResNet [33]. The output of this layer is fed to a fully-connected layer and an output layer for the 2 classes activated by a softmax function where the BCE loss is used to train the classifier.

3.5. Stage 5: Post-Processing

The output mask from stage 2 will be of size $H \times W$ where each of the elements corresponding to the target object will have values between 0.5 and 1, and less than 0.5 otherwise. The center (x_c, y_c) of the target object location can be calculated as the average location of these elements weighted by the confidence, which is the output of the sigmoid function, as follows:

$$x_c = \frac{\sum_{k=1}^{K} \sigma(x_k) x_k}{\sum_{k=1}^{K} \sigma(x_k)}, \quad y_c = \frac{\sum_{k=1}^{K} \sigma(y_k) y_k}{\sum_{k=1}^{K} \sigma(y_k)}. \tag{7}$$

K is the number of elements where the confidence $\sigma(x, y)$ is higher than the threshold, and $\sigma(x_k) = \sigma(y_k) = \sigma(x_k, y_k)$. The bounding box width and height can be calculated as: $width = \beta_x \sigma_x$ and $height = \beta_y \sigma_y$, where β_x and β_y are factors that are learned during the training of the backbone network. σ_x and σ_y are the weighted standard deviation of locations corresponding to elements with values higher than the threshold. The output of stage 3 is used to track the object as it moves from one frame to another, by averaging the movement that is estimated within the predicted bounding box. Finally, the classifier is used to decide whether the frame can be classified as a frame that contains the target object or not.

4. EXPERIMENTS AND RESULTS

4.1. Dataset

We evaluated our model on an ultrasound dataset created by researchers at UC San Diego (UCSD) Health and Jacobs School of Engineering. The experimental procedures involving human subjects described in this paper were approved by the Institutional Review Board (IRB) at UCSD (IRB No. 171154). The target object in the dataset is the Vagus nerve in the mid- and upper-cervical regions of the neck. The scans span different locations on the neck generating several different fields of view of the neck representing a sonographer who is looking for the Vagus nerve within the neck. The dataset contained 6,368 scans from 3 different subjects, and contained scans from both the left and right side of the neck. The Vagus nerve shape, location and surrounding anatomical structures varies greatly within subjects and across subjects. Even a slight movement of the probe can make it challenging for sonographers to re-identfy and localize it due to the high variability of neck anatomy visualized with mediolateral or cepahlo-caudal scanning along the cervical neck [34]. In aggregate, nerve detection with the variable anatomy dataset, provides a substantial challenge for which we will test our proposed method and verify its effectiveness.

4.2. Implementation and Setup

We conducted 2 experiments to test the performance and robustness of our framework. The 1st experiment was designed to test the accuracy of the proposed framework in detecting and localizing the Vagus nerve. The dataset was divided into individual scans and split into a 64:16:20 ratio for training, validation, and testing, respectively (the scans were resized to 256x256). The backbone network in stage 2 was optimized using stochastic gradient descent (SGD) with a learning rate = 10^{-3}, momentum = 0.9, weight decay of 10^{-3}, batch size = 16, and trained for 200 epochs. The optical flow network in stage 3 was trained stage-wise [31] on the Chairs dataset [32], then on the Things3D dataset [35]. The training used the Adam optimizer and learning rates 1e-4, 5e-5, and 4e-5. The 2nd experiment was designed to test the framework ability to use the optical flow network in stage 3 effectively to track the movement of the Vagus nerve accurately. The proposed work was implemented in Pytorch [36].

4.3. Evaluation and Results

To evaluate the detection and localization accuracy of the proposed framework, we used the average precision and recall of localization. This evaluation metric considers a detection as true positive when a certain localization threshold is met, otherwise that detection is considered a false positive. This is the main metric used to evaluate object detection algorithms [20, 21]. The localization threshold is based on the intersection over union (IoU) metric. The proposed framework achieved an average precision of 94.4% and average recall of 97.2% for an IoU threshold of 0.5. This surpasses the average precision and recall performance of both YOLOv4 and EfficientDet - d3. For YOLOv4, the average precision was 93.2% and the average recall was 97.2%, while for EfficientDet - d3 the average precision was 94.1% and the average recall was 96.8%. For tracking, the framework achieved state-of-the art performance in ultrasound object tracking, where the mean error is less than 0.75 mm at real-time speeds of 30 fps when tracking the Vagus nerve across frames. Tracking results are summarized in Table 1, where the performance of the proposed framework is presented together with several prominent tracking algorithms. These algorithms were designed and trained to track objects in ultrasound and their performance was tested on Challenge on Liver Ultrasound Tracking (CLUST) [37]. For the classification of target frames, the classifier in stage 4 precision and recall are 93.01% and 86.25%, respectively. Finally, the framework achieves high localization precision where more than 95% of the true positive detections are within 1.5 mm from the ground truth in both the lateral and axial directions. Predicted location offset from the ground truth is shown as a heatmap in Fig. 3 for both the object detection and tracking predictions. To contextualize the framework localization effectiveness, it is worth noting that the axis span in Fig. 3 is smaller than the sides of the bounding boxes in Fig. 1.

Table 1: Mean and standard deviation of tracking error in mm for the different ultrasound tracking methods. Autonomous detection of objects and real-time capabilities are highlighted.

Method	Mean	σ	Detection?	Real-Time?
Shepard [6]	0.72	1.25	No	No
Williamson [9]	0.74	1.03	No	Semi (8 fps)
Gomariz [12]	1.34	2.57	No	Yes (105 fps)
Makhinya [8]	1.44	2.8	No	Yes (20 fps)
Proposed	0.73	1.23	Yes	Yes (30 fps)

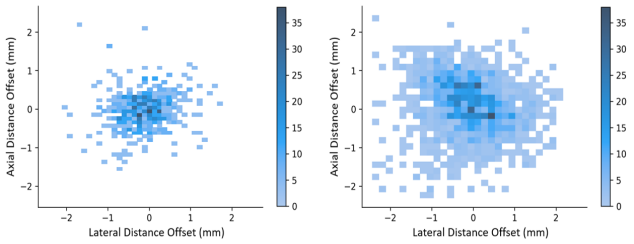

Fig. 3: The lateral and axial distance offset of the true positive detections from the ground truth in millimeters. Left: Tracking heatmap of offsets. Right: Detection heatmap of offsets.

5. CONCLUSION

In this paper, we presented a real-time framework based on a weakly trained segmentation network and an optical flow estimation network to detect and track a target object in ultrasound scans. We tested its performance by using it to detect and track the Vagus nerve in ultrasound scans of the neck, where a complex anatomical structure is present. Results of the testing demonstrated that such a framework is accurate and is able to autonomously detect a target object and track its location throughout a scanning session to assist sonographers in performing diagnostic and therapeutic procedures.

6. REFERENCES

[1] A. L. Klibanov and J. A. Hossack, "Ultrasound in radiology: from anatomic, functional, molecular imaging to drug delivery and image-guided therapy," *Investigative radiology*, vol. 50, no. 9, pp. 657, 2015.

[2] D. L. Miller, N. B. Smith, M. R. Bailey, *et al.*, "Overview of therapeutic ultrasound applications and safety considerations," *Journal of ultrasound in medicine*, vol. 31, no. 4, 2012.

[3] E. Tegnander and S. Eik-Nes, "The examiner's ultrasound experience has a significant impact on the detection rate of congenital heart defects at the second-trimester fetal examination," *Ultrasound in Obstetrics and Gynecology*, vol. 28, no. 1, 2006.

[4] P. J. Keall, G. S. Mageras, J. M. Balter, *et al.*, "The management of respiratory motion in radiation oncology report of aapm task group 76 a," *Medical physics*, vol. 33, no. 10, 2006.

[5] A. Giachetti, "Matching techniques to compute image motion," *Image and Vision Computing*, vol. 18, no. 3, pp. 247–260, 2000.

[6] A. J. Shepard, B. Wang, T. KF. Foo, *et al.*, "A block matching based approach with multiple simultaneous templates for the real-time 2d ultrasound tracking of liver vessels," *Medical physics*, vol. 44, no. 11, pp. 5889–5900, 2017.

[7] D. C. Wang, R. Klatzky, B. Wu, G. Weller, A. R. Sampson, and G. D. Stetten, "Fully automated common carotid artery and internal jugular vein identification and tracking using b-mode ultrasound," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 6, 2009.

[8] Maxim Makhinya and Orcun Goksel, "Motion tracking in 2d ultrasound using vessel models and robust optic-flow," *Proceedings of MICCAI CLUST*, vol. 20, pp. 20–27, 2015.

[9] T. Williamson, W. Cheung, S. K. Roberts, *et al.*, "Ultrasound-based liver tracking utilizing a hybrid template/optical flow approach," *International journal of computer assisted radiology and surgery*, vol. 13, no. 10, pp. 1605–1615, 2018.

[10] E. Ozkan, C. Tanner, M. Kastelic, *et al.*, "Robust motion tracking in liver from 2d ultrasound images using supporters," *International journal of computer assisted radiology and surgery*, vol. 12, no. 6, 2017.

[11] S. Bharadwaj and M. Almekkawy, "Deep learning based motion tracking of ultrasound image sequences," in *2020 IEEE International Ultrasonics Symposium (IUS)*, 2020, pp. 1–4.

[12] A. Gomariz, W. Li, E. Ozkan, *et al.*, "Siamese networks with location prior for landmark tracking in liver ultrasound sequences," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE, 2019, pp. 1757–1760.

[13] S. Bharadwaj and M. Almekkawy, "Faster search algorithm for speckle tracking in ultrasound images," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, 2020, pp. 2142–2146.

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

[15] Zhen Ma, João Manuel RS Tavares, and RM Natal Jorge, "A review on the current segmentation algorithms for medical images," in *Proceedings of the 1st International Conference on Imaging Theory and Applications (IMAGAPP)*, 2009.

[16] Xiangbin Liu, Liping Song, Shuai Liu, and Yudong Zhang, "A review of deep-learning-based medical image segmentation methods," *Sustainability*, vol. 13, no. 3, pp. 1224, 2021.

[17] I. Lerman, R. Hauger, L. Sorkin, *et al.*, "Noninvasive transcutaneous vagus nerve stimulation decreases whole blood culture-derived cytokines and chemokines: a randomized, blinded, healthy control pilot trial," *Neuromodulation: Technology at the Neural Interface*, vol. 19, no. 3, pp. 283–290, 2016.

[18] A. P. Mourdoukoutas, D. Q. Truong, D. K. Adair, *et al.*, "High-resolution multi-scale computational model for non-invasive cervical vagus nerve stimulation," *Neuromodulation: Technology at the Neural Interface*, vol. 21, no. 3, pp. 261–268, 2018.

[19] M. M. Ottaviani, L. Wright, T. Dawood, *et al.*, "In vivo recordings from the human vagus nerve using ultrasound-guided microneurography," *The Journal of Physiology*, vol. 598, no. 17, 2020.

[20] A. Bochkovskiy, C. Wang, and H. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[21] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020.

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015.

[23] S. Lee, J. S. Bae, H. Kim, *et al.*, "Liver lesion detection from weakly-labeled multi-phase ct volumes with a grouped single shot multibox detector," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2018, pp. 693–701.

[24] I. Castiglioni, L. Rundo, M. Codari, *et al.*, "Ai applications to medical images: From machine learning to deep learning," *Physica Medica*, vol. 83, pp. 9–24, 2021.

[25] A. F. Al-Battal, Y. Gong, L. Xu, *et al.*, "A cnn segmentation-based approach to object detection and tracking in ultrasound scans with application to the vagus nerve detection," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2021.

[26] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr, "Fully-convolutional siamese networks for object tracking," in *European conference on computer vision*. Springer, 2016, pp. 850–865.

[27] P. Simard, D. Steinkraus, and J. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, 2003, vol. 2, pp. 958–958.

[28] Zhishuai Zhang, Siyuan Qiao, Cihang Xie, Wei Shen, Bo Wang, and Alan L. Yuille, "Single-shot object detection with enriched semantics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[29] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.

[30] A. P. Zijdenbos, B. M. Dawant, R. A. Margolin, and A. C. Palmer, "Morphometric analysis of white matter lesions in mr images: method and validation," *IEEE Transactions on Medical Imaging*, vol. 13, no. 4, pp. 716–724, 1994.

[31] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy, "Liteflownet: A lightweight convolutional neural network for optical flow estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8981–8989.

[32] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[33] 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*, 2016, pp. 770–778.

[34] A. T. Ahuja, *Diagnostic Ultrasound: Head and Neck*, Diagnostic Ultrasound. W. B. Saunders, 2019.

[35] N. Mayer, E. Ilg, P. Hausser, *et al.*, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4040–4048.

[36] A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019.

[37] V. De Luca, J. Banerjee, A. Hallack, *et al.*, "Evaluation of 2d and 3d ultrasound tracking algorithms and impact on ultrasound-guided liver radiotherapy margins," *Medical physics*, vol. 45, no. 11, 2018.