*University of Ottawa*

# A Real-time and Automatic Ultrasound-Enhanced Multimodal Second Language Training System: A Deep Learning Approach

*By*

**Mohammad Hamed Mozaffari Maaref**

*A thesis submitted in partial fulfillment of the requirements for the Doctorate in Philosophy degree in Electrical and Computer Engineering*

*Ottawa-Carleton Institute for Electrical and Computer Engineering*
*School of Electrical Engineering and Computer Science*
*University of Ottawa*

*Winter 2020*

# Abstract

The critical role of language pronunciation in communicative competence is significant, especially for second language learners. Despite renewed awareness of the importance of articulation, it remains a challenge for instructors to handle the pronunciation needs of language learners. There are relatively scarce pedagogical tools for pronunciation teaching and learning, such as inefficient, traditional pronunciation instructions like listening and repeating. Recently, electronic visual feedback (EVF) systems (e.g., medical ultrasound imaging) have been exploited in new approaches in such a way that they could be effectively incorporated in a range of teaching and learning contexts. Evaluation of ultrasound-enhanced methods for pronunciation training, such as multimodal methods, has asserted that visualizing articulator's system as biofeedback to language learners might improve the efficiency of articulation learning. Despite the recent successful usage of multimodal techniques for pronunciation training, manual works and human manipulation are inevitable in many stages of those systems. Furthermore, recognizing tongue shape in noisy and low-contrast ultrasound images is a challenging job, especially for non-expert users in real-time applications. On the other hand, our user study revealed that users could not perceive the placement of their tongue inside the mouth comfortably just by watching pre-recorded videos.

Machine learning is a subset of Artificial Intelligence (AI), where machines can learn by experiencing and acquiring skills without human involvement. Inspired by the functionality of the human brain, deep artificial neural networks learn from large amounts of data to perform a task repeatedly. Deep learning-based methods in many computer vision tasks have emerged as the dominant paradigm in recent years. Deep learning methods are powerful in automatic learning of a new job, while unlike traditional image processing methods, they are capable of dealing with many challenges such as object occlusion, transformation variant, and background artifacts. In this dissertation, we implemented a guided language pronunciation training system, benefits from the strengths of deep learning techniques. Our modular system attempts to provide a fully automatic and real-time language pronunciation training tool using ultrasound-enhanced augmented reality. Qualitatively and quantitatively assessments indicate an exceptional performance for our system in terms of flexibility, generalization, robustness, and autonomy outperformed previous techniques. Using our ultrasound-enhanced system, a language learner can observe her/his tongue movements during real-time speech, superimposed on her/his face automatically.

# Acknowledgments

I am indebted to many people who played a significant role in the completion of this dissertation. First and foremost, I express my gratitude to my lovely wife, Niloufar, for her tender mercies in granting me light and knowledge in pursuing this research. I am grateful to have had the opportunity of researching while living in relative comfort. I recognize that this is not a blessing that everyone receives.

I would like to express my sincere appreciation to Prof. Won–Sook Lee for her constant support during my doctorate studies and her patience, motivation, and immense knowledge. Her critical comments and insightful feedbacks were essential for any steps towards the progress of my research. Her guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D. study.

I also would like to thank all my friends and lab members, Shuangyue Wen, Chandho Kim, Aminur Ratul, Jiawei Li, Hamed Mozafari, Maryam Tavakolelahi, Chao Sun, to name a few for their in-depth knowledge, support, and constructive feedback. Last but not least, I would like to thank my parents (Javad & Zohreh), my mother-in-law (Maryam), and sister-in-law (Shaghayegh) for supporting me spiritually throughout writing this thesis and my life in general.

# List of Contents

# List of Figures

# List of Tables



# List of Equations

# Chapter 1 Introduction

In this thesis dissertation, we focus on the application of supervised deep learning techniques for the tasks of object detection, tracking, and segmentation. Our datasets are real-time videos from different ultrasound machines and cameras. For further evaluation, we also employed standard semantic segmentation benchmarks. The outcome of each study contributes to a module of our guided language (L2) pronunciation training system. We thoroughly answered the questions and difficulties discussed in our proposed research plan.

## 1.1 Motivation

One of the critical aspects of the second language (L2) acquisition as an integral part of communication skills is pronunciation. From the perspective of a listener, it is often the first indication of a language learner's linguistic abilities *(Bird et al., 2018)*. Pronunciation directly affects many social interaction skills of a speaker, such as communicative competence, performance, and self-confidence. Furthermore, previous studies revealed that other aspects of L2 learning, such as word learning, can be improved by accurate pronunciation *(Johnson et al., 2018)*.

Besides the importance of L2 pronunciation, it is one of the most challenging skills to master for adult learners *(Abel et al., 2015)* in traditional classroom settings. There is often no explicit pronunciation instruction for language learners because of limited class time and lack of knowledge of effective pronunciation teaching and learning methods *(Abel et al., 2015)*. Standard practice for a language learner outside of the classroom is to imitate a native speaker's utterances in front of a mirror, limited to lip and jaw movements, along with hearing of recorded acoustic data.

Without any visual feedback of a native speaker and lack of awareness of how sounds are being articulated, it is difficult for language learners to improve their skills *(Abel et al., 2015)*, especially in cases where the target sounds are not easily visible *(Bird et al., 2018)*. The positions and movements of the tongue, especially all but the most anterior part, cannot be seen in the traditional approach of listening and repeating word's pronunciations *(Bliss, Burton, et al., 2017)*. Language learners can only have proprioceptive feedback of their tongue location depends on practicing sounds (vowels,

liquids, or others) and the amount of contact their tongue makes with the teeth, gums, and palate *(Wilson et al., 2006)*.

Many previous investigations in the literature of L2 pronunciation acquisition and ability have been focused on acoustic studies dealing with the sound that is produced and infer the articulation that created the sound *(Wilson et al., 2006)*. Nevertheless, both acoustic and articulatory studies are undoubtedly valuable tools for understanding the progress of an L2 learner. The latter one can often give a more accurate picture of the actions performed by the pronunciation learner while it looks directly at the articulators (e.g., the tongue, the lips, and the jaw) *(Wilson et al., 2006)*. Employing acoustic data alone might jeopardize the understanding of L2 learners in mapping the acoustic information onto articulatory movements. Having seen the articulators directly by learners, they can probably improve their pronunciation by the perception of the articulatory adjustments *(Wilson et al., 2006)*. Therefore, an effective way of pronunciation teaching and learning includes listening and repeating using both acoustic and articulatory information.

Enhancing ultrasound video frames by highlighting the tongue dorsum region improved the trend of pronunciation training of language learners with a more straightforward interpretation of real-time ultrasound data *(Abel et al., 2015; Yamane et al., 2015; Yuen et al., 2011)*. Furthermore, extracted tongue contour curves provide teachers and language researchers valuable information for quantitatively comparisons and detailed investigation studies *(Laporte & Ménard, 2018)*. It is noteworthy to mention that tongue contour extraction is accomplished after tongue region segmentation using an image processing technique such as skeletonizing or just keeping the top pixels of the tongue region (see Figure 3-1 for a sample of tongue surface region and extracted contour).

Typically, during tongue data acquisition, ultrasound probe beneath the user's chin images tongue surface in midsagittal or coronal view *(Slud et al., 2002)* in real-time. Midsagittal view of the tongue in ultrasound data is usually adapted instead of a coronal view for illustration of tongue region, as it displays relative backness, height, and the slope of various areas of the tongue. Tongue dorsum can be seen in this view as a thick, long, bright, and continuous region due to the tissue-air reflection of ultrasound signal by the air around the tongue (see Figure 2-4 and Figure 3-1). This thick white bright region is irrelevant, and the tongue surface is the gradient from white to the black area at the lower edge *(Stone, 2005)*.

Although the tongue contour region can be seen in ultrasound data, there is no reliable solid structure as a reference in ultrasound data to locate the tongue position and

interpret its gestures *(Stone, 2005)*. Furthermore, due to the noise characteristic of ultrasound images with a low-contrast property, it is an even more laborious task for non-expert users to follow the tongue movements, especially in real-time applications *(Bliss et al., 2016)*. Few previous multimodal studies proposed manual highlighting tongue regions with different colors *(Yamane et al., 2015)*, which is not applicable for automatic and real-time applications. In this project, we introduced several fully automatic and real-time image segmentation methods from deep learning literature to track the surface of the tongue in video frames (as a continuous high-lighted thick region). As mentioned before, from our user study, language learners can follow automatically delineated tongue contours more convenient than just watching raw ultrasound data in real-time.

Recent technology-assisted language learning methods, such as multimodal approaches using ultrasound imaging, have been successfully employed for language pronunciation teaching and training, providing visual feedback of learner's whole tongue movements and gestures *(Abel et al., 2015; Antolik et al., 2019; B. Bernhardt et al., 2005; Bird et al., 2018; Gick et al., 2008; Hueber, 2013; Yamane et al., 2015)*. However, this technology is still far from commercializing for use in language training institutes. There are several difficulties for the current pronunciation training systems, while main issues are:

- Current multimodal systems are not real-time, where users can not observe their tongue and face together. Instead, pre-recorded videos from native speakers are illustrated for the user after manual pre- and post-processing stages. Interactive, automatic, and real-time systems can help language learners to perceive their tongue placement in their mouth during an L2 learning session *(Yamane et al., 2015)*.
- Non-expert language learners need a reference to understand the tongue location in real-time while in current systems, this guidance is ignored. Manual coloring the whole tongue is not a generic alternative for this issue. Tracking palate as a reference also requires sharp images of the mouth while palate position is not visible in all frames.
- In almost all the previous studies, the head of language learners should be fixed using stabilizers such as helmets. Therefore, users do not have enough flexibility during training sessions. In cases that quantitative study is the goal of a research ultrasound probe should also be fixed using designed helmets that decrease the flexibility of users.

- Current methods are not repeatable, and they work only in specific system setup and customized datasets *(B. Bernhardt et al., 2005)*. For instance, a technique that works well on one dataset is not generalizable for other datasets. Similarly, previous methods are not applicable to all ultrasound systems.

The motivation of this research is to provide a real-time and fully automatic modular system to address the difficulties of previous systems by utilizing state-of-the-art deep learning techniques. Our pronunciation training system comprises of several modules that can be used efficiently for language pronunciation teaching and learning. The system can work on all types of ultrasound machines in different system setup and environment.

## 1.2 Objectives and Challenges

We observe many hurdles for deploying a real-time and automatic system for the second language (L2) pronunciation training. To address these difficulties, we implemented several novel approaches for each part of our system. Specifically, we focused and proposed solution separately on different computer vision applications such as:

1) **Automatic and real-time tongue image segmentation:** the interpretation of noisy ultrasound data capture from the tongue is not an easy task, especially when for real-time application. For this reason, one solution is to delineate and track the tongue surface automatically from each ultrasound frame in real-time *(Bliss et al., 2016)*. Current methods of tongue contour tracking require manual or semi-automatic data initialization and enhancements *(K. Xu, Gábor Csapó, et al., 2016)*. These methods are not usable for real-time applications where their performance highly depends on the quality of the data, manual initialization, and limited to the memory and computational availability. Few recently utilized deep learning techniques for tongue contour segmentation *(Laporte & Ménard, 2018)* are not powerful enough as a general method applicable to any ultrasound tongue dataset when few datasets are publicly available.
2) **Automatic and real-time object tracking:** previous multimodal ultrasound guided L2 pronunciation training systems use stabilizers to fix the place of the language learner's head and the ultrasound probe. The reason is that these systems do not work in real-time, and there is no room for calculation of transformations for the overlapping process. In these systems, ultrasound tongue data are transformed manually on the user's face *(Abel et al., 2015; Bird et al., 2018; Yamane et al., 2015)*. For this reason, there should be an automatic and real-time

tracking system to determine the optimum placement of the ultrasound data on the face of a language learner.

3) **Augmented Reality:** Tracking the tongue surface and ultrasound probe location should be accomplished both simultaneously. Tongue contours are highlighted (segmented) on ultrasound raw frames. Then, the overlapped frame of the ultrasound tongue and corresponding segmented contour is superimposed on the video frame of the user's face using the transformation data determined by an automatic tracking module. Therefore, all techniques should work together with the same framerate, which is a challenging task for multimodal approaches in L2 pronunciation learning systems. For the accurate quantitative evaluation of the previous tongue contour tracking methods, one selected frame should be frozen to calculate statistical information *(Abel et al., 2015; Bird et al., 2018)* as a post-processing stage. Our automatic and real-time system enables researchers to acquire this information during training sessions. Therefore, ultrasound data and segmented tongue contour should be automatically augmented on the face view of the user in real-time.

Besides these challenges of the project, there are other difficulties that we investigated in our proposed system, including occlusion for the tracking method, the generalization of the system such as overfitting and underfitting in deep learning models, tongue datasets with similar distributions, to name a few. It is also noteworthy to mention that one of our primary goals was to create a system accessible for all researchers as affordable as possible.

## 1.3 Contributions

Artificial intelligence (AI) is a branch of computer science when machines can do tasks that typically require human knowledge *(Hamet & Tremblay, 2017)*. Machine learning is a subset of AI, where devices can learn by experience and acquire skills without human involvement. Inspired by the functionality of the human brain, artificial neural networks learn from large amounts of data to perform a task repeatedly. Deep learning algorithms are artificial neural networks with various (deep) layers similar to the human brain structure *(LeCun, Bengio, & Hinton, 2015)*. Deep learning-based methods and their applications in image processing literature, such as object detection and image segmentation have been a research hotspot in recent years. Deep learning methods are powerful in automatic learning a new task. Unlike traditional image processing methods, they are capable of dealing with many challenges such as object occlusions, transformations, and background artifacts *(Chen, Papandreou, Kokkinos, Murphy, &*

*Yuille, 2014; Guo et al., 2016; Zhao, Zheng, Xu, & Wu, 2018)*. We proposed several deep learning architectures to address each part of our pronunciation system as follows.

### 1.3.1 Automatic and Real-time Tongue Contour Tracking

After extensive investigation of the field, we proposed novel deep learning models for automatic tongue contour tracking in real-time ultrasound video frames. We called those methods as sU-NET, sDeepLabV3, BowNet, wBowNet, and IrisNet. All models utilize the power of the convolutional neural network in different architectures. Following state-of-the-art methods in semantic segmentation literature, IrisNet and BowNet models are specifically designed for tongue contour segmentation with high accuracy and performance using graphical processing units (GPUs) while in lack of this facility, sU-NET can work well on Central Processing Unit (CPU) power. Our contribution to this project is not limited to network implementation. We also designed a new convolutional module inspired by the peripheral vision ability of the human eye. Usage of dilated convolutions for better contextual information acquisition for the problem of tongue contour tracking is another novelty of our models. Unlike previous ultrasound tongue contour tracking, IrisNet is trained and tested on datasets with two binary classes of foreground (tongue) and background instead of a gray-level ground-truth label. Note that in semantic segmentation, each target object has a unique label. Therefore, using grayscale images for ground-truth labels might be considered as 255 different target objects. On the other hand, at this end, predictions also will be grayscale.

Our experimental results indicate that our approaches significantly outperform the state-of-the-art image segmentation methods. From an extensive dataset developing projects, we created two challenging annotated ultrasound tongue datasets. In a separate study, the generalization ability of the deep learning method is investigated using the domain adaptation approach. We introduced a new term to the transfer learning literature called a knowledge-balanced point. The overfitting problem is not an issue for our last proposed models anymore due to using data augmentation, optimized architectures, usage of state-of-the-art convolutional layers, and following training methods in the semantic segmentation field.

### 1.3.2 Automatic and Real-time Ultrasound Probe Tracking

Tracking a user's face has been employed as a reference in different computer vision studies. In our preliminary experiments, we used the face profile of a user as a reference similarly. Our results indicate a high dependency of the system on the quality and characteristics of users' faces. Furthermore, different face view provides different features.

For this reason, instead of tracking the face profile, we used automatic tracking of the ultrasound probe. Information from the tracking method helps us to calculate transformations (calibration data) required for real-time augmenting ultrasound frames on the face-side view of the user.

Following the landmark tracking field by deep learning techniques *(Y. Wu & Ji, 2019)*, we designed a convolutional neural network named ProbeNet for automatic tracking of the ultrasound probe in real-time. To train ProbeNet independent from face profiles, we used simulated human face data combined with real data. As another contribution, we designed a 3D printable device named UltraChin for making the system fully independent from environment properties as well as higher accuracy for the language quantitative evaluation study using our language pronunciation training system. One electromagnetic tracking sensor attached to UltraChin helps us to calculate our ProbeNet accuracy.

### 1.3.3 An Ultrasound-Enhanced L2 Pronunciation Training System

We deployed two different pronunciation training systems with different setups employing UltraChin, ProbeNet, BowNet, and IrisNet, while a language learner could observe her/his tongue in real-time on her/his face. All components of the system work automatically together simultaneously without any manipulation and data enhancement. The experimental results of our language training system revealed its compatibility with the data from any ultrasound machines due to the excellent generalization ability of IrisNet and BowNet models. Different ultrasound probe types with different shapes can be utilized with our 3D printable UltraChin device. To assess our system in terms of usability, we conducted a user study experiment inviting several language learners to test our system in real-time.

### 1.3.4 Novel Idea for Ultrasound Tongue Tracking Field

Besides previous contributions, we also proposed a new idea of training neural networks to track specific points on the tongue surface. Our experimental results revealed that this idea could be a promising technique for the future of automatic and real-time tongue contour tracking in ultrasound video data. It is noteworthy that this idea is not part of our language pronunciation system, and for this reason, we present our qualitative results. Although the idea of tracking points on the surface of the tongue using a deep learning approach works well, an extensive quantitative study should be accomplished. One of the significant benefits of this technique, besides improvement of performance, simplicity, and robustness, is that almost all ultrasound tongue datasets in the Linguistics

department accompany by point annotated data of the tongue surface. Using this idea, we can train a deep learning model with a considerable number of annotated data (like ImageNet in image classification area) which might be a significant pace for this literature.

## 1.4 Structure of the Thesis

The rest of this document is organized into the following chapters:

- Chapter 2 provides a brief background review, including ultrasound technology and deep learning methods for object tracking and semantic image segmentation.
- Chapter 3 presents a detailed literature review of the linguistics application of ultrasound imaging. It provides previous in-depth techniques of tongue contour tracking and application of ultrasound in L2 acquisition.
- Chapter 4 explains our created datasets for training deep learning models in this project. Details of each dataset are also described in the experimental results.
- Chapter 5 elaborates on our methodology by describing each contribution to the two main modules of our system.
- Chapter 6 presents evaluation studies for each module and comparison assessments with similar techniques. A user study of our system is presented in this chapter.
- Chapter 7 concludes the thesis. It also outlines the directions for future work.

## 1.5 Scientific Articles from this project

From this project, we submitted and published several scientific papers as follows:
**Journal articles:**

1) (Published) Hamed Mozaffari, M., & Lee, W.-S. (2019). Domain adaptation for ultrasound tongue contour extraction using transfer learning: A deep learning approach. *The Journal of the Acoustical Society of America*, *146*(5), EL431–EL437. https://doi.org/10.1121/1.5133665
2) (Published) Mozaffari, M. H., Sankoff, D., & Lee, W.-S. (2019). Ultrasound tongue contour extraction using BowNet network: A deep learning approach. The Acoustical Society of America, 178th Proceeding of Meetings on Acoustics (Journal of ASA POMA), vol. 39

3) (Published) Mozaffari M. H. and Lee W.-S., “Real-time Ultrasound-enhanced Multimodal Imaging of Tongue using 3D Printable Stabilizer System: A Deep Learning Approach”, Canadian Acoustics, vol. 48, no. 1, Mar. 2020.
4) (Accepted for Publication) Journal of Second Language Pronunciation: Mozaffari, M. Hamed, et al. Real-time and Fully Automatic Ultrasound Multimodal Visual Biofeedback for Second Language Teaching and Learning: A Deep Learning Approach.
5) (Submitted) Mozaffari, M. Hamed, et al. "IrisNet: Deep Learning for Automatic and Real-time Tongue Contour Tracking in Ultrasound Video Data using Peripheral Vision." Journal of IET Image Processing and Arxiv preprint arXiv:1911.03972 (2019).
6) (Submitted) Mozaffari, M. Hamed, et al. BowNet and IrisNet: Two Deep Learning Methods for Tongue Contour Tracking in Ultrasound Videos. Methods, Elsevier, 2020.

**Other Presentations:**

1) (Published) M. Hamed Mozaffari and Won-Sook Lee, Ultrasound Tongue Contour Extraction using Dilated Convolutional Neural Network. IEEE International Conference on Bioinformatics and Biomedicine 2019.
2) (Published) Mozaffari, M. Hamed, et al. "Guided Learning of Pronunciation by Visualizing Tongue Articulation in Ultrasound Image Sequences." 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA). IEEE, 2018.
3) (Published) Mozaffari, M. Hamed, et al. "Real-time automatic tongue contour tracking in ultrasound video for guided pronunciation training." Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. Vol. 1. 2019.
4) (Submitted to ICIP2020) Peripheral Vision for Semantic Segmentation without Transfer Learning.
5) (Submitted to MICCAI2020) Deep Learning for Automatic Tracking of Tongue Surface in Real-time Ultrasound Videos, Landmarks instead of Contours
6) (Published abstract) Mozaffari, M. H., Sankoff, D., & Lee, W.-S. Transfer learning for ultrasound tongue contour extraction with different domains. *The Journal of the Acoustical Society of America 146, EL431 (2019);* https://doi.org/10.1121/1.5133665

7) (Published abstract) Mozaffari, M. H., Sankoff, D., & Lee, W.-S. (2019). BowNet: Dilated convolutional neural network for ultrasound tongue contour extraction. The Journal of the Acoustical Society of America, 146(4), 2940–2941. https://doi.org/10.1121/1.5137212
8) (Published in press) Mozaffari, M. H., Sankoff, D., & Lee, W.-S. (2019). Artificial Intelligence for Automatic Tracking of the Tongue in Real-time Ultrasound Data. 178th ASA Meeting, San Diego, CA. https://acoustics.org/3pba4-artificial-intelligence-for-automatic-tracking-of-the-tongue-in-real-time-ultrasound-data-mhamed-mozaffari/

**Posters Presentations:**

1) Mozaffari, M. H., Sankoff, D., & Lee, W.-S. Transfer learning for ultrasound tongue contour extraction with different domains. The Journal of the Acoustical Society of America 146, EL431 (2019); https://doi.org/10.1121/1.5133665
2) Mozaffari, M. H., Sankoff, D., & Lee, W.-S. (2019). BowNet: Dilated convolutional neural network for ultrasound tongue contour extraction. The Journal of the Acoustical Society of America, 146(4), 2940–2941. https://doi.org/10.1121/1.5137212

**Relevant research publications during the Ph.D. Project:**

1) (Published) (Calibration Calculations) Mozaffari, Mohammad Hamed, and Won-Sook Lee. "Freehand 3-D ultrasound imaging: a systematic review." Ultrasound in medicine & biology 43.10 (2017): 2099-2124.
2) (Published) (Image Segmentation Basics) Mozaffari, Mohammad Hamed, and Won-Sook Lee. "Convergent heterogeneous particle swarm optimization algorithm for multilevel image thresholding segmentation." IET Image Processing 11.8 (2017): 605-619.
3) (Arxiv) (Medical Image Segmentation Basics) Mozaffari, Mohammad Hamed, and WonSook Lee. "3D Ultrasound image segmentation: A Survey." arXiv preprint arXiv:1611.09811 (2016).
4) (Arxiv) (Medical Image Segmentation Basics) Mozaffari, Mohammad Hamed, and Won-Sook Lee. "Multilevel thresholding segmentation of T2 weighted brain MRI images using convergent heterogeneous particle swarm optimization." arXiv preprint arXiv:1605.04806 (2016).
5) (Published, Winner of University of Ottawa poster competition 2018) Mozaffari, M. Hamed, et al. "Real-time automatic tongue contour tracking in ultrasound

video for guided pronunciation training." Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. 2019.

6) (BioArxiv) (Dilated convolutions for other applications) Ratul, A. R., Mozaffari, M. H., Lee, W. S., & Parimbelli, E. (2019). Skin Lesions Classification Using Deep Learning Based on Dilated Convolution. bioRxiv, 860700.
7) (BioArxiv) (Transfer Learning for other applications) Ratul, M. A. R., Mozaffari, M. H., Parimbelli, E., & Lee, W. (2019). Atrous Convolution with Transfer Learning for Skin Lesions Classification. BioRxiv, 746388.

# Chapter 2 Literature Review

## 2.1 Ultrasound Technology

### 2.1.1 Physics of Ultrasound

Ultrasound waves (or vibrations) are generated by tiny but fast push-pull actions of air, which are produced in a transducer (probe). Frequency in hertz (Hz) or cycles per second of these kinds of waves is higher than the auditory perception of the human. In terms of physical characteristics, all types of vibration as a sound wave are known as acoustic waves, and they can be classified into different classes depending on their frequency ranges. For instance, in medical applications, acoustic waves with a rate of vibration from 20 kilohertz (kHz) to 50 megahertz (MHz) are used for imaging purposes. It is noteworthy to mention that the human ability for hearing sound waves is only for vibration rates between 20 Hz to 20 kHz. The frequency of sound waves used in medical studies is higher than human hearing ability, and for this reason, they referred to as ultrasound.

Piezoelectric crystals are a kind of material that can be deformed quickly by applying electrical signals to them (in the form of electrical current). This characteristic of Piezoelectric crystals is a reversible process, and they produce electrical signals by applying a mechanical force to them. Therefore, they can be used as both transmitter and receiver of ultrasound signals. This property of piezoelectric crystals makes acoustic waves in medical ultrasound imaging. Several crystals are mounted onto a hand-held case, and they are triggered to work in a consecutive manner utilizing customized electronics, consist of amplifiers, converters, and filters.

When electrical pulses are applied to the crystals, according to a specific timeline, corresponding crystals produce ultrasound waves one after the other. The ultrasound wave propagates through tissues, and depending on the type of each tissue, decays or is absorbed (converted to heat energy), refracted (wave direction is changed due to the different propagation speeds of each tissue), passes through medium (is diffracted in the form of diverging and spreading out from the source crystal), scattered (because of interaction with small tissue structures like bubbles), and reflected toward crystals. A comprehensive survey on the fundamentals of ultrasound technology can be found in studies by *(Allan, P., Weston, Michael, Baxter, 2011; Prince, Jerry L., Links, 2015)*.

The speed of sound, c, is related to the frequency, f, and the wavelength, λ, by Equation 2-1. If the speed of sound can be considered as a constant value (the speed of

sound varies depending medium type), researchers can change wavelength of ultrasound waves by changing its frequency. Therefore, there is a trade-off between frequency and visualization depth for medical ultrasound imaging, such that higher frequencies cause more wave energy absorption and faster attenuation by tissues. Consequently, the wave can only penetrate shorter distances (near depth with skin as a reference).

$$c = f \times \lambda \qquad \text{Equation 2-1}$$

Therefore, decreasing the frequency of acoustic wave allows researchers to see deeper objects but with less accurate resolution. Nevertheless, a more in-depth scanning needs more reflection time, and it causes a slower scan rate, which is a problem in real-time applications *(Allan, P., Weston, Michael, Baxter, 2011)*.

### 2.1.2 Ultrasound Image Acquisition

The reflected ultrasound wave is converted into an electrical signal, and it is prepared for illustration as an image (Figure 2-1 shows the conversion of ultrasound wave to electrical signals). By knowing speed of sound wave, c, which is close to 1540 m/s for most soft tissues, with five percent tolerance, and the time, t, that takes for ultrasound wave to propagate through the body organs and return to the crystal (twice the depth), it is possible to calculate the distant (depth) of reflection point from the crystal, d, using Equation 2-2.

$$2 \times d = \mathrm{c} \times \mathrm{t} \qquad \text{Equation 2-2}$$

In this way, received electrical pulses by one crystal is converted into distances and form a line of intensity data known as A-mode (amplitude mode). The position and intensity of each pixel in that line are calculated from the delay and amplitude of each received pulse, respectively. Although intensity is a critical factor for the reconstruction of ultrasound signals, there are many other aspects and details for the conversion of signals into an image known as ultrasound beamforming techniques *(Luchies & Byram, 2018)*. For instance, an ultrasound beam converges to one or several focal points depends on the technologies and settings of the ultrasound system. Those points are similar and behave like the focus point in a camera.

In the medical ultrasound imaging field, several A-mode data from a line of mounted crystals in a probe are concatenated to form a two-dimensional (2D) image called B-mode (brightness mode). Figure 2-2 shows a simple illustration of B-mode image reconstruction from several A-mode signals. It is noteworthy to mention that because one

A-mode image is capture in ultrasound sweeping, the left side of the reconstructed image is slightly older than the right side.

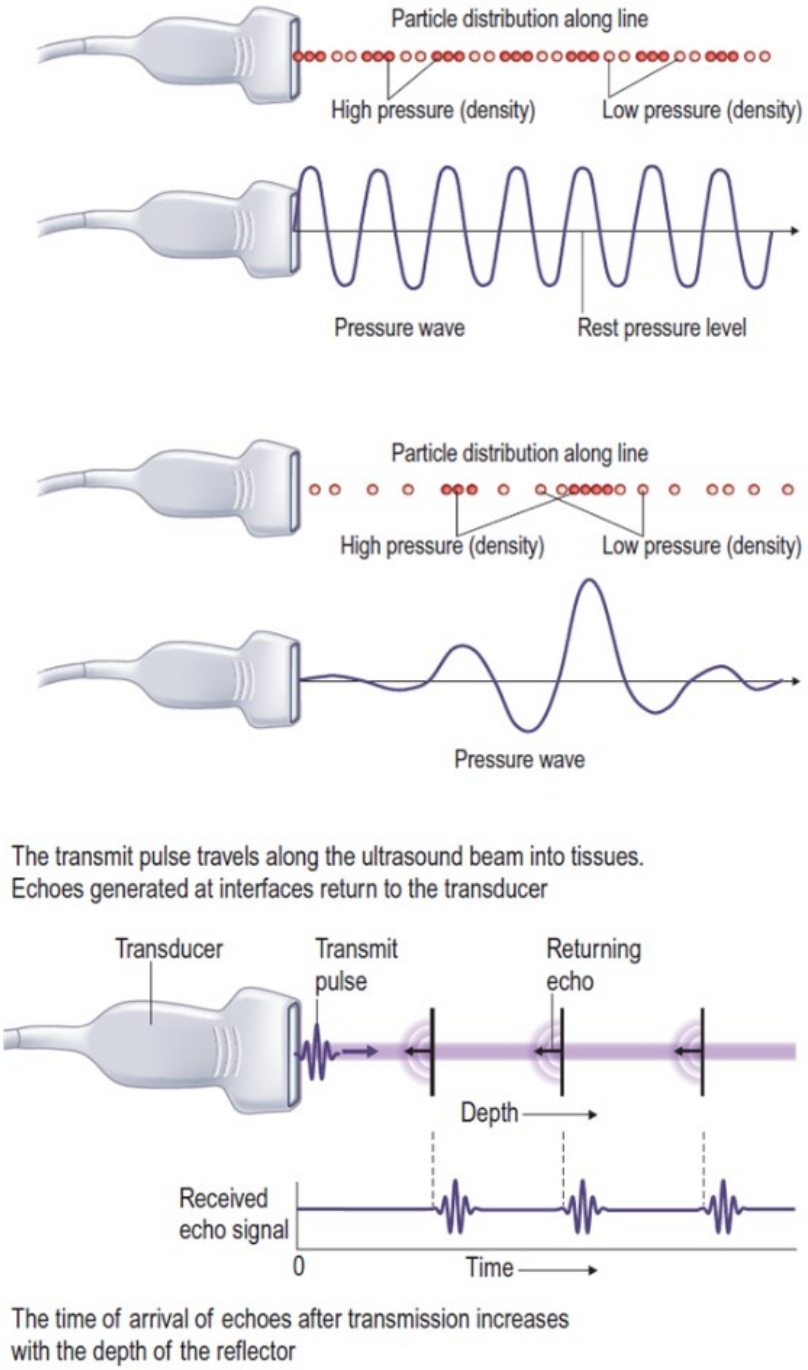



Figure 2-1. A sample of pulse and echo cycle, fundamental for image construction in ultrasound (Allan, P., Weston, Michael, Baxter, 2011).

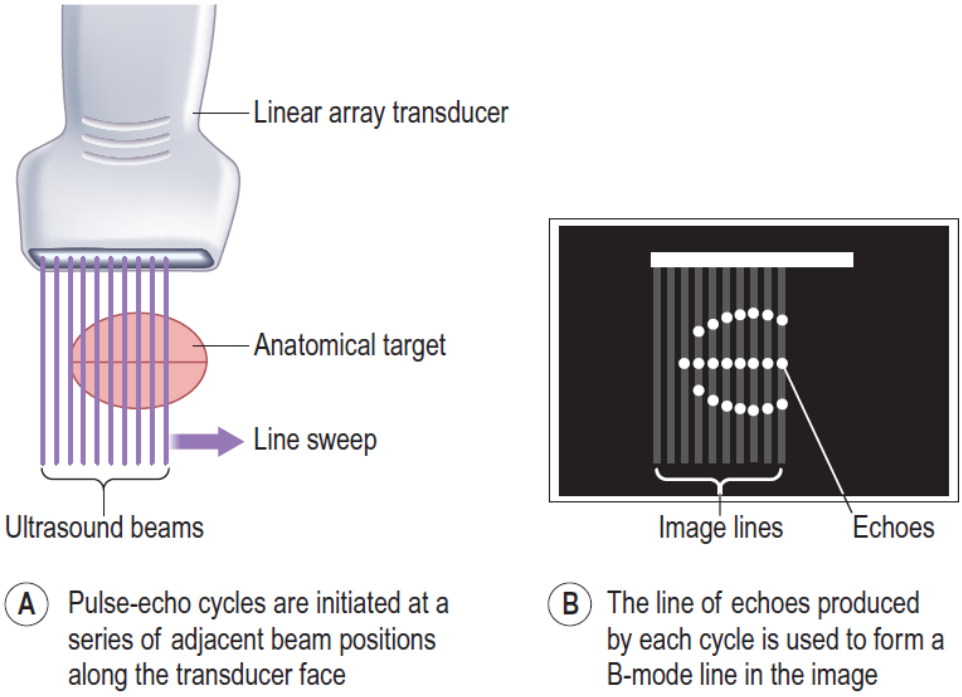



Figure 2-2. B-mode image formation in ultrasound imaging technology. Concatenation of several consecutive ultrasound pulses and echoes in the form of several A-mode lines construct a 2D B-mode image (Allan, P., Weston, Michael, Baxter, 2011).

Because of differences between the characteristics of human tissues in different organs, muscles, fat regions, blood fluid, and air, a significant portion of ultrasound signals reflects from the borders of these regions. Therefore, physicians can see different structures and organs in ultrasound B-mode images with abundant information. However, the interpretation of ultrasound images is often required trained personnel. For this reason, it is a challenging task for non-expert people to recognize detailed information in ultrasound images, more complicated than understanding the data from other medical imaging modalities such as magnetic resonance imaging (MRI), or computed tomography (CT).

In this section, we only briefly explain the core procedure of B-mode image acquisition. Nevertheless, in ultrasound devices, there are many other challenges that engineers should tackle to acquire a clear image for illustration. A flow diagram of core modules and processes of forming a B-mode image in an ultrasound device can be seen in Figure 2-3. In a higher level of data acquisition technique, ultrasound data can be captured in 3D formats. A comprehensive survey of three-dimensional (3D) ultrasound image acquisition techniques has published recently *(Mohammad Hamed Mozaffari & Lee, 2017)*.

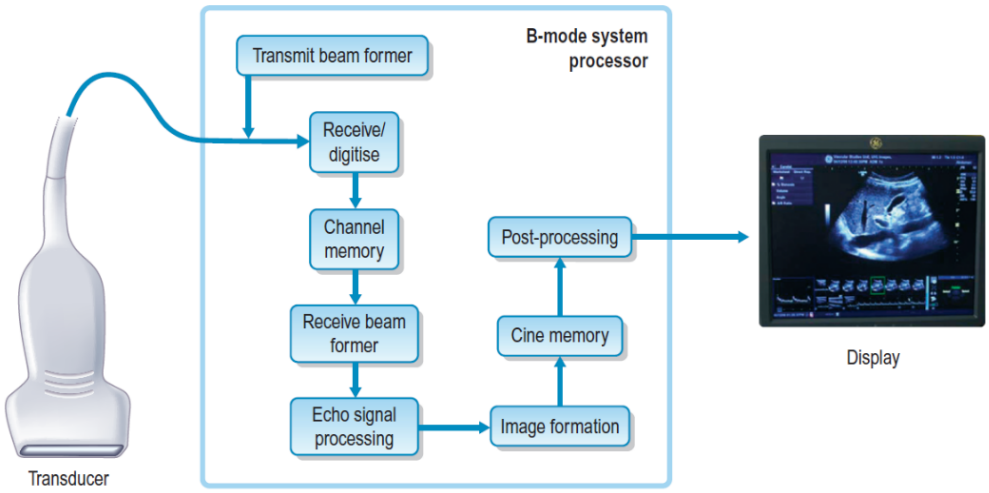



Figure 2-3. Main modules of an ultrasound device for the acquisition of one B-mode image (Allan, P., Weston, Michael, Baxter, 2011).

### 2.1.3 Ultrasound Image Analysis

In this section, we explain some basic terms in ultrasound literature, which are essential for the understanding of other sections. As mentioned before, ultrasound images are noisy and with low contrast characteristics. The reason for low-quality images can be traced back to the physics and the properties of ultrasound waves in encountering different media such as boundaries between human tissues. Followings are the main terms for ultrasound wave properties when passing through a medium (Allan, P., Weston, Michael, Baxter, 2011), each of which might impact ultrasound image quality and contrast in different ways.

**Diffraction:** when an ultrasound wave passes from its source through a medium or passes through a small hole, the direction of the beam is changed, and the ultrasound beam spreads out as a diverging shape. The spreading pattern is highly dependent on the size and shape of ultrasound crystals relative to the wavelength of the sound wave.

**Interference:** waves can be overlapped when they pass through the propagating medium. This overlapping appears as a superposition of two or more waves. Any wave has three characteristics in 2D space: amplitude, which is the maximum value of the wave in each cycle, frequency which has an inverse relation with wavelength, and phase, which is the difference between zero axes as a reference and starting point of the wave in each cycle. If the phases of two or more waves are the same, those waves are added (constructive interference). Otherwise, they cancel each other (destructive interference).

**Reflection:** a reflection of the ultrasound wave happens whenever the beam strikes the boundary of two media where there is a difference between the density of the two media. The acoustic impedance of each medium is how readily tissue particles move about their rest position under the influence of the wave pressure. To calculate the amount of reflection, we need to calculate the difference of acoustic impedance between two media. Acoustic impedance for a medium is directly related to the density of the medium and the speed of sound in that medium. Therefore, higher impedance means higher density and slower sound velocity. For instance, gases like air usually have a high value of acoustic impedance, and they reflect almost all parts of the sound wave.

**Scattering:** when the ultrasound wave reaches a tissue with a smaller size than the wavelength, and different variations in its acoustic impedance, some of the wave energy is scattered in many directions.

**Refraction:** as the ultrasound beam is reflected from a boundary of two media, portions of that beam crosses the boundary with a different angle than its previous direction. This change of direction with a different angle is called ultrasound refraction. If the beam direction is precisely perpendicular to the boundary, there is no beam bending or refraction.

**Absorption and attenuation:** the energy of ultrasound is converted from a vibrational form into random vibrational heat after passing through tissue. Hence, the wave amplitude reduces after traveling from the source. The higher the frequency of ultrasound waves, the more the reduction of its amplitude. Both absorption and attenuation also depend on the type of medium (human tissue) and other effects such as reflection and scattering.

**Speckle noise in ultrasound images:** while the ultrasound beam is passing through a tissue region, a considerable number of small discontinuities reflect small portions of the beam to the transducer. A coherent summation of each signal from many scatters are detected in the ultrasound transducer by interfering with other echoes. The result is a high amount of noise, which in the image appears as speckle patterns. This kind of interference is detrimental to image quality. However, in some applications such as 3D ultrasound, reconstruction can be used to find the relation between images in terms of position and orientation for freehand scanning *(Mohammad Hamed Mozaffari & Lee, 2017)*. Speckle noise significantly degrades the image quality and complicates the understanding of fine details in ultrasound images *(Rabbani et al., 2008)*. Speckle noise has a random and deterministic nature as it is formed from backscattered echoes of randomly or coherently distributed scatters in the tissues.

Statistical properties of speckle noise are highly dependent on the density and the spatial distribution of the scatterers *(Thijssen, 2003)*. Many statistical distributions have been utilized to simulate the scattering of speckles in ultrasound images *(Insana et al., 1985)* while the Rayleigh distribution was used in the particular case of a large number of randomly located scatterers in a study by *(Osman & Kaftandjian, 2017)*. Various methods have been proposed to reduce speckle noise, such as filtering in spatial, temporal, or frequency domain *(Gai et al., 2018)* by studying different noise distributions. However, lack of an accurate distribution and computational complexity have made this topic as an open area for research. In clinical applications, researchers control the effect of speckle noise by changing ultrasound gain in the region of interest.

Summation of all the effects mentioned above makes it a complicated task to reconstruct an ultrasound image from echo signals. Moreover, ultrasound images usually contain granular noise patters called the speckle noise that may have negatively affect image interpretation.

### 2.1.4 Tongue Structures in Sagittal View

The anatomical structure of the neck-lower chin area is complex, and for a precise interpretation and understanding of B-mode images of the tongue, knowing human anatomy is required. Ultrasound images of the tongue usually contain other visible structures than only the tongue itself, some of which are introduced below (see Figure 2-4).

**The velum:** Velum (known as soft palate) is the root of the nasal cavity (an essential region for velar consonant sounds in speech, for more details refer to the International Phonetic Alphabet (IPA) such as *(Lawson, E., Stuart-Smith, 2019)*). In detail, the lingua-velar contact allows a packet of air to pass into the velum and reflect from the air above it *(Stone, 2005)*.

**The Palate:** Palate is the roof of the mouth where it separates the oral cavity from the nasal cavity. Usually, the palate cannot be seen in ultrasound images because when the tongue is at rest position, there is a hallow area between tongue surface and palate skin. Therefore, ultrasound waves cannot travel through the air between them. However, it is valuable to track the palate and consider that in ultrasound images as a reference structure *(Stone, 2005)*. The importance of tracking palate is that for an accurate measurement, the head should be the reference instead of the Jaw in many studies, and the palate has a fixed spatial location with respect to the head. During swallowing of

water or food, the tongue can approximate the place of the palate by touching its surface. Therefore, the palate might be visible in one or a group of consecutive ultrasound frames. **Hyoid bone:** this U-shape bone, near to the root of the tongue on the neck, is an essential structure for the tongue movements. Like the Jaw, the hyoid bone reflects the ultrasound wave and results in a black region in ultrasound images. Sometimes, the reflection of hyoid bone might form a bright area in ultrasound images *(Stone, 2005)*.

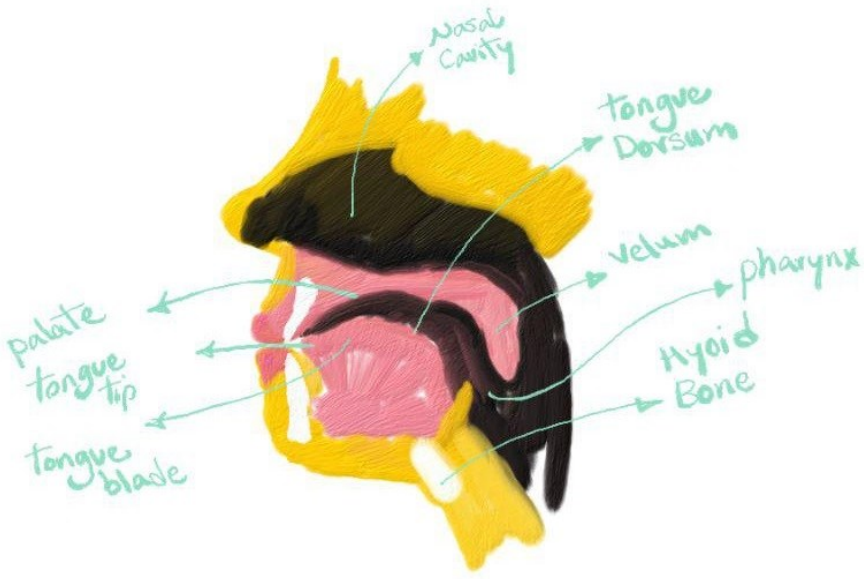



Figure 2-4. A cross-section of mid-sagittal view of the tongue and neck-lower chin region[1].

**Tongue wholistic:** The whole tongue is from Pharynx to tip. Because of ultrasound reflection and refraction, the tongue tip or root cannot be seen clearly in ultrasound images. Ideally, the ultrasound wave travels through various human tissues and organs until it reaches to a tissue-air interface, in which almost all the energy is absorbed or reflected. For this reason, when the strong reflection occurs, structures beyond it could rarely be imaged. Moreover, if an earlier interface causes a considerable sound reflection, the tongue surface itself might not be imaged well. In the area under the tip of the tongue, there could be shadow regions due to the Jaw and the hollow of air.

During recording one ultrasound image of the tongue, the tissue-air interface reflects almost all the power of the sound wave. Therefore, the deeper structures (such as the Palate) beyond the tissue-air surfaces (with high acoustic impedances) cannot be captured, resulting in black regions in the reconstructed ultrasound image. Furthermore,

[1] This drawing was made by the author using dry pastel.

the lower Jaw (called Mandible) and hyoid bones, which refract the sound before it penetrates the tongue surface, have a similar effect on ultrasound images *(Stone, 2005)*. The resultant of those structures in ultrasound images is called an acoustic shadow, which depends on the characteristics of the bone density, appears on the lower part of the image (around the root of the tongue).

Acoustic shadow can obscure some critical parts or movement patterns of the tongue. A common practice to alleviate this issue is to push the transducer further up, press tightly against the Jaw. This act usually makes it clear for the tongue tip to show up in ultrasound images, although it causes a significant deformation of the posterior tongue *(Stone, 2005)*. In applications such as tongue modeling where a highly accurate image reconstruction is required or in a precise quantitative analysis of the tongue surface, a considerable amount of error in this alternative is inevitable.

### 2.1.5 Ultrasound for Tongue Imaging

It is difficult to observe and study tongue function due to its deep position within the oral cavity that causes inaccessibility for most instruments. Furthermore, any sensing devices within the mouth must be unaffected by moisture and temperature. They must not cause any difficulty for motions of the tongue. Ultrasound technology can be an alternative for this issue, and it can indirectly illustrate tongue gestures. In general, medical ultrasound imaging is a non-invasive, relatively affordable, and portable technology, which made it possible for visualization of the human tongue in real-time.

Ultrasound imaging plays a crucial role in many real-time applications due to the capacity of scanning with relatively higher video frame rates (scan rate of 60 Hz to 120 Hz), such as for heart monitoring compared to other medical image modalities. For this reason, ultrasound technology can be utilized for the illustration of rapid dynamics and complex shapes of tongue in real-time. When setting the frequency of the ultrasound probe, it is suggested that typically, a frequency of 7.5 to 17 MHz would be suitable for examining the tongue *(Allan, P., Weston, Michael, Baxter, 2011)*. This selection of frequency depends on the size of the submandibular region of patients. For instance, a 9 MHz ultrasound probe could be suitable for most adults, while a 12 MHz probe is usually suggested for children or slim adults *(Allan, P., Weston, Michael, Baxter, 2011)*.

The ultrasound gel between probe and skin is crucial for the examination process, for the medical ultrasound wave has such high frequencies that it is either attenuated quickly or reflected entirely in the air before sending out from the probe. The best position for speech examinations is with the stretched patient's neck, and some additional force

ought to be applied to make a solid contact between probe and skin, with the chin in the midline. An experiment for one subject should take around 10 minutes, to allow the subject to find the best location and adjust the image to the best quality, and then pronounce the given set of words. According to our experiments, the tongue surface (dorsum) appears around 7 to 9 centimeters from the probe head in a typical ultrasound display (depend on the ultrasound probe type). For this reason, for better illustration of the tongue dorsum, focal points of the ultrasound probe and gain compensation amplifier should be set for that distance range. Drinking water before each examination can slightly improve the quality of video frames.

About 40 years ago, one-dimensional ultrasound was first used effectively for illustration of one point at a time on the tongue's surface *(Kelsey et al., 1969)*. The two-dimensional ultrasound image (B-mode settings for mid-sagittal or coronal view) has been employed in speech research since 25 years ago *(Sonies et al., 1981)*. Nevertheless, ultrasound imaging technology has used recently for visualizing the speech articulators in speech research. Due to the recent development of this technology with higher image quality and greater affordability, it became an essential tool for pedagogical use in the acquisition of L2 pronunciation *(Bird et al., 2018; Wilson et al., 2006)*.

## 2.2 Deep Learning Techniques

### 2.2.1 A Brief Introduction to Deep Learning

Artificial intelligence (AI) is an essential branch of science whose aim is to train computers, tools, and robots to repeat a task similar or even better than human performance. Machine learning (ML) is one of the approaches to achieve AI, and in recent years deep learning (DL) methods, which are new ML techniques, have been focused on considerable efforts of researchers. Deep learning impacts successfully on so many complicated problems that scientists could not solve them efficiently before, especially in the image processing and computer vision realm.

Applications of machine learning in our daily life can be seen in many fields such as web search techniques, commercial recommendations in e-commerce websites, diagnosis of diseases in medical science, helping researchers in Astronomy to find new celestial objects, assisting mathematicians to solve complicated equations, natural language processing that enables computers to communicate with human without being explicitly programmed in advance, predicting the potential of drug molecules and mutations in genetics and bioinformatics, and in general in any place where we are looking to

distinguish some patterns in a database (also known as pattern recognition and data mining).

The development of ML methods has accelerated since 2006 when deep learning techniques emerged as a promising area for research, especially when *(Geoffrey E. Hinton et al., 2006)* proposed a novel deep structured learning (deep belief network (DBN)). Many breakthroughs have happened in deep learning areas in recent years. For instance, the research group led by Hinton *(Hansson, 2002)* won the competition of ImageNet 2012 by introducing a novel deep learning approach with significantly better results than previous methods, introducing of generative adversarial networks (GANs) by *(Goodfellow et al., 2014)* for generating new images from a database, AlphaGo algorithm successfully won the game Go *(Silver et al., 2016)*, or recently proposed ideas such as one-shot (or few-shot) learning *(Fei-Fei et al., 2006; Vinyals et al., 2016)* and capsule neural networks *(Sabour et al., 2017)*.

The two main common characteristics of deep learning methods are multiple layers of non-linear architectures[1] and more feature abstraction in successively higher *(L. Deng, 2014)*. Deep learning popularity owes to advances in sensors and data digitization technologies, which enable scientists to access big databases for training. Moreover, the development of big data analysis techniques helps to solve the over-fitting problem in training data partially (the over-fitting problem is explained in later sections). Other reasons are recent developments in computer algorithms such as new efficient optimization algorithms, advances in computing processing using graphical or tensor processing units (GPUs or TPUs) *(Jouppi et al., 2017)* as well as decrease in cloud computing costs, emergence of popular deep learning competitions such as ImageNet and Kaggle, and the pre-training procedure instead of randomized initialization for network parameters.

Machine learning techniques, designed before the deep learning era, have shallow-structure architectures, which usually consist of one or more layers of non-linear feature transformations. Examples of those shallow architectures can be enumerated as Gaussian mixture models (GMMs) *(Reynolds et al., 2000)*, linear or non-linear dynamical systems *(Svensson & Schön, 2017)*, conditional random fields (CRFs) *(Zheng et al., 2015)*, maximum entropy models *(Och & Ney, 2001)*, support vector machines (SVMs) *(Drucker et al., 1999)*, logistic regression *(Stoltzfus, 2011)*, and multilayer perceptrons (MLPs)

[1] Many problems in engineering and science are non-linear though we can approximate them as linear functions. To model these non-linearities, a simulation method like neural networks should be capable of coping with non-linearities. Adding non-linear functions after a linear layer is a method to solve this problem.

*(Gori & Scarselli, 1998)* with a single hidden layer including extreme learning machines (ELMs) *(L. Deng, 2014)*.

Although traditional ML methods have been used successfully and efficiently to address many premier pattern recognition problems, a significant improvement has always required for these methods to get better results for complex real-world applications. For instance, traditional techniques cannot adequately solve image processing problems such as image segmentation, image registration, classification of big data, real-time object recognition, and tracking in video frames. The ability of the human to solve these complicated problems encourages researchers to search for a better solution with simulating different models mimicking the pattern recognition and perception mechanism of the human brain.

Deep hierarchical models with many layers (each of which followed by a non-linear function) have shown better performance and accuracy than previous shallow-structured models. An excellent example of this successful simulation of human brain is that researchers used traditional artificial neural networks (see Figure 2-5) but this time with many hidden layers with newly defined non-linearities among each layer (referred to as deep neural networks (DNNs)) *(L. Deng, 2014; W. Liu et al., 2017)*. DNN is a multilayer neural network with many fully connected hidden layers, which is initialized by unsupervised or a supervised pre-trained network[1].

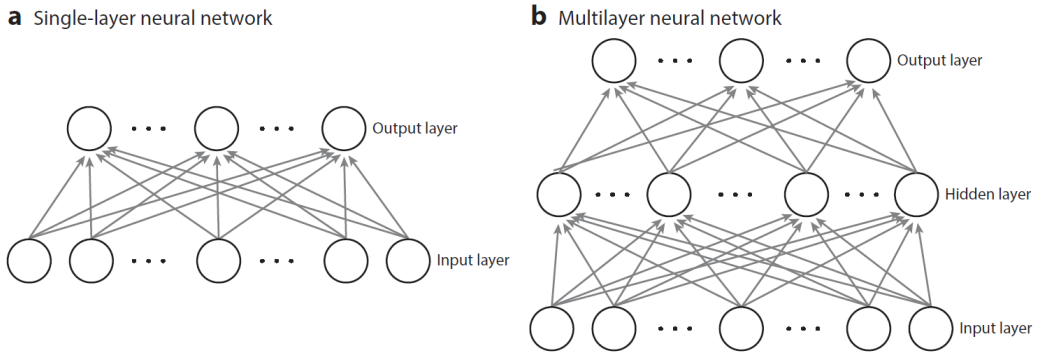



Figure 2-5. Architecture of two feed-forward neural networks *(Shen et al., 2017)*. a) without hidden layer, b) with one hidden layer.

Backpropagation (BP) has been used for learning the parameters of neural networks since 1980. Although it is a popular method for this task, its performance is significant only for convex problems, while in encountering non-convex problems with

[1] All major deep learning techniques and fundamental concepts are described in the next sections.

many local optima, it performs poorly with unacceptable outcomes. Deep neural network applications might be considered as non-convex problems, especially when parameters of networks initialize randomly. Training of such a network by the BP method provides unsatisfactory results, and scientists sense a need for the development of novel ideas for improving BP performance for deep models.

Various studies empirically reveal that using parameters of a pre-trained model instead of random initialization causes significantly better results with the alleviation of BP and optimization difficulties. In general, using more hidden layers with many neurons and initialization utilizing pre-trained models for a deep neural network reduces the chance of trapping in poor local optima. Other factors can assist deep learning models in finding a better solution, including better network design with efficient non-linearities like Rectified Linear Units (original or leaky)*(B. Xu et al., 2015)*, utilizing better optimization algorithm such as Stochastic Gradient Descent (SGD) *(Ruder, 2016)* and Adam optimization method *(Kingma & Ba, 2014)*, to name few of them.

In a typical taxonomy, deep learning architectures might be classified into five classes: I) supervised learning methods (labeled and unlabeled training dataset are provided), II) unsupervised learning techniques (only unlabeled training dataset is available), III) semi-supervised deep networks as a combination of supervised and unsupervised approaches (Usually model is trained in two different steps), and IV) Reinforcement learning (training approach is based on action and rewards). There are endless numbers of deep architectures in the context of deep learning, and each of which has been exploited for many applications. Here, we summarize popular and fundamental models from the literature, and we just briefly explain the general idea of each one.

Deep belief networks (DBNs) *(Bengio et al., 2007; Geoffrey E. Hinton et al., 2006)* inspired the whole machine learning field by a new learning algorithm that can find a great set of parameters for multi-layer non-linear models by utilizing a small amount of labeled data during the whole training period. There are two types of probabilistic models according to the theory of probability and statistics. For predicting the class $y$ of the given observation $x$, the generative model calculates the joint distribution $p(x, y)$, and the discriminative model calculates the conditional distributions $p(y|x)$. So, because the base of deep learning techniques is on probabilistic models, in another taxonomy, they can be classified as deep generative models (are trained by unsupervised methods), deep discriminative models (trained by supervised techniques), and combination of them as deep hybrid models *(J. Xu et al., 2015)*. In the following sections, these classes of deep learning models are explained in more detail.

### 2.2.2 Generative Models

As mentioned before, a generative model (sometimes is considered as an unsupervised learning model) is a statistical model comprises of a joint probability distribution of observed and unobserved data. There are many generative models, such as the Hidden Markov model, Naive Bayesian model, Gaussian mixture model, and neural networks. Here we briefly explain some recent, popular neural network generative models. A summary of all models which we are explaining can be seen in Figure 2-6.

Boltzmann machine (BM) *(Geoffrey E. Hinton & Sejnowski, 1986)* is an asymmetric network with neuron-like units that make stochastic decisions of on or off like a transistor in digital electronics. This model consists of a layer of visible units and a layer of hidden units such that there are no visible-visible or hidden-hidden connections between them. Restricted Boltzmann machines (RBMs) *(Fischer & Igel, 2012)* are widely used in the deep learning field. An RBM network consists is a variant of Boltzmann machines (BMs), and they have been used to generate stochastic models of artificial neural networks that can learn the probability distribution with respect to their inputs. BMs can be interpreted as neural networks with stochastic processing units, connected bidirectionally (see Figure 2-6 as an example).

Some popular applications of RBMs are topic modeling, dimensionality reduction, collaborative filtering, data classification, and feature learning. An RBM might be used to encode the data, then applied as an unsupervised learning model or as a generative model for regression and classification applications. Using Bayes law to calculate the joint distribution of the visible and hidden units, an RBM can also be used as a discriminative model. In general, RBMs are utilized as feature extractors in the pre-training process for classification tasks. The hidden and visible variables in RBM are not mutually independent, and training of a BM can be difficult and time-consuming in practice. RBMs were proposed to overcome this problem by restrictions on the network topology. However, due to the existence of other problems such as the intractability of the log-likelihood gradient, it could not alleviate the training issue significantly. The RBM has become prominent since the publication of *(Geoffrey E. Hinton et al., 2006)* work when they constructed the deep belief networks (DBNs) by stacking a bank of RBMs.

DBN model comprises multiple layers of stochastic hidden variables. There are two kinds of layers with undirected and symmetric connections in the top (last blocks) and the directed connections in the lower layers (first blocks). Different layers of RBMs in a DBN are trained sequentially. The lower RBMs are trained first, then the higher ones. Training of a DBN can be divided into two stages: the pre-training stage, which is a down-

up direction unsupervised training, and the fine-tuning stage as an up-down supervised learning to adjust the parameters of the network further. After the pre-training stage, the trained parameters are used to initialize the deep network. Then, utilizing a supervised learning algorithm such as backpropagation, the deep learning model is fine-tuned with significantly better outcomes. A Deep Boltzmann Machine (DBM) *(Salakhutdinov & Larochelle, 2010)* is a kind of DBN but with many layers of hidden variables with just an undirected graphical model. Like DBN, DBM is trained using stacked RBMs but with the difference that the input is doubled for the lower-level RBMs.

Generative Adversarial Networks (GANs) *(Goodfellow et al., 2014)* are deep neural network models comprised of two networks, the generator and the discriminator that set up a kind of minimax game between each other to train the whole network. The generator creates samples from a noise distribution, and it aims to fool the discriminator that classifies those samples as real or fake. One ability of GAN network models is that it can differentiate fake and real data, and for computer vision applications, it needs to add more ability to the network to create clear and real images. Variational autoencoders (VAEs), a new variant of generative models, solve this problem by providing higher resolution created images.

Autoencoder (AE) *(Geoffrey E. Hinton & Zemel, 1994)* is another type of neural network that has been used as an unsupervised learning algorithm for dimensionality reduction. Parameters of a trained AE might be used for other supervised learning models to improve their accuracy and solve the over-fitting problem. The AE is trained to encode the input into some representation so that the input can be reconstructed from that representation. A common AE is a one-hidden-layer feed-forward neural network, and the main difference between an MLP and an AE is that the AE aims to reconstruct the input, while the purpose of the MLP is to predict the target values with specific inputs.

Autoencoders usually comprise three kinds of layers: an input layer, which gets the input as a vector, hidden layer or layers (sometimes called Deep Autoencoder), which represents the transformed feature, and the output layer, which generates a matching between input and output data. Similar to that for the DBNs, the training process for an AE can also be divided into two stages: training stage: after random initialization of parameters, feed-forward propagation is performed, and the purpose is to obtain the output value as an estimated value of the input. The error will be backpropagated through the network using the chain rule method to update the weights. In the fine-tuning stage, one supervised learning method is adopted, and the gradient descent algorithm adjusts the parameters at each layer. In the case of deep AEs due to the vanishing gradient

problem and other difficulties, initialization of the network is usually done by using parameters of a pre-trained network such as a DBN pre-trained model.

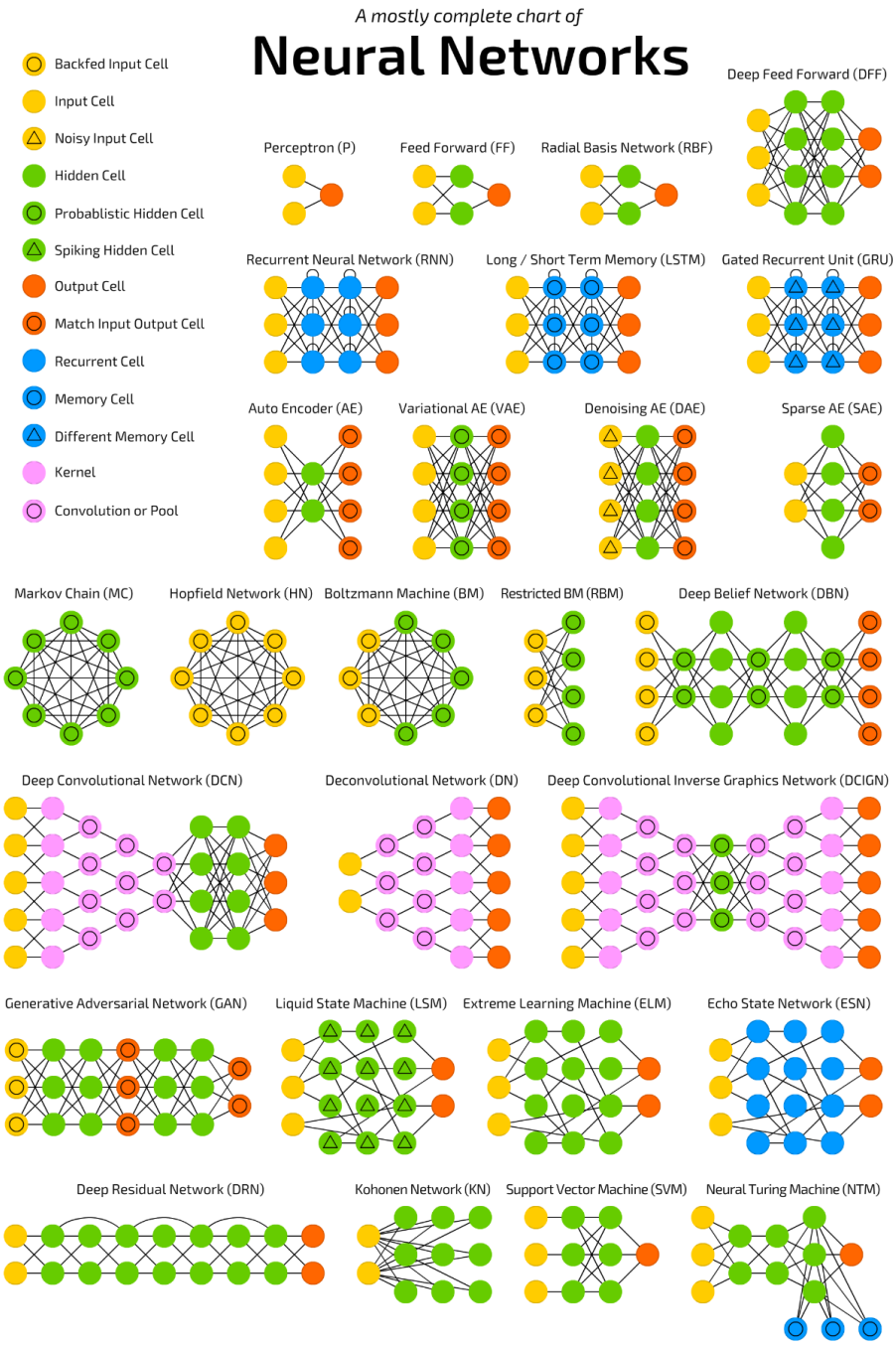



Figure 2-6. The general idea of neural network architectures in terms of their hidden layers and cells *(Kojouharov, n.d.)*

There are various popular AE architectures in the literature, and we enumerate some here. For instance, denoising the traditional AEs (DAEs) have been proposed by *(Vincent & Larochelle, 2008)*. Through the training process, the DAE can recover the noise-free version of the training data, which implies enhanced robustness. Another architecture was developed based on stacked layers of DAEs *(Vincent et al., 2010)* consisting of multiple layers of sparse AEs such that the outputs of each layer are wired to the inputs of the successive layer. Other models are AEs with sparsity like k-sparse AE *(Makhzani & Frey, 2013)* that keeps only the highest $k$ activations in the hidden layers. In contractive autoencoders (CAEs) *(Rifai et al., 2011)*, a well-selected penalty term is added to the standard cost function in the reconstruction stage. It is employed to penalize the sensitivity of the features with respect to the inputs. A separable deep autoencoder (SDAE) was proposed by *(Sun et al., 2016)*, which is used to deal with the unseen noise estimation. Deep autoencoder *(G. E. Hinton, 2006)* is another unsupervised discriminative DNN which provides output similar to its input without any class labels. Applying denoising criteria for training, a deep AE is also a generative model.

Variational autoencoders (VAEs) *(Kingma & Welling, 2014)* work like a standard AE with encoder (convolutional layers[1]) and decoder (deconvolutional layers), but in encoding stage, the VAE is constrained to generate latent vectors that roughly follow a Gaussian unit distribution along with two vectors of mean and standard deviation (SD). So, in the decoder part, VAE can decode a sampled latent vector into image (like a generator in GANs) efficiently.

### 2.2.3 Discriminative Models

The aim of discriminative models (also known as conditional models because of conditional probability) is to model the relationship of unobserved samples with observed data. In contrast with generative models, which require joint distribution for problems such as classification and regression, discriminative models can yield superior performance in these applications. There are many discriminative models in the machine learning field, such as support vector machines, conditional random fields, random forests, linear and logistic regressions, neural networks, and so many more. Here we briefly survey well-known architectures of neural networks, where they are trained in a supervised fashion.

In computer science, artificial neural networks (ANNs) (see Figure 2-5) are machine learning models that are inspired by the structure of biological neural networks of living creatures. They can learn to perform tasks by observing many examples. For instance, in

[1] Convolutional architecture will be explained in discriminative models.

object recognition tasks, one ANN model can learn to identify an object in an image by analyzing many labeled data. Similar to generative deep learning models, the ANNs are a collection of connected layers, each of which comprises many units called neurons. The accuracy of ANNs has been significantly increased after the advent of deep convolutional neural networks (DCNNs) *(Krizhevsky et al., 2012)*. The DCNNs are deep ANNs that have been used for many applications with impressive results, including machine vision and image processing problems. DCNN models usually consist of many consecutive convolutional and pooling layers (often Max-pooling), as well as some non-linearities (such as ReLU) between layers. In contrast to DBN or ANN, in which all nodes in architecture are connected to the other nodes, in convolutional layers, each layer is connected to only a smaller portion of the next layer units. This type of network connection (applying a kernel on data to modify weights instead of weight multiplication between layers) gives DCNN a kind of receptive field which, along with pooling layers, enables the network to be faster and invariant to transformations *(Simonyan & Zisserman, 2015)*.

Recurrent Neural Network (RNN) *(Ayoub & Al Osman, 2019; Graves et al., 2013)* is mainly designed for handling sequences, and it can recognize patterns in sequences of data emanating from recording devices such as microphones, keyboards, cameras, internet, and stock markets. Popular RNN architectures such as long short term memory (LSTM) *(Sak et al., 2014)* has a short memory, and they can take input in real-time and also work on data that they have perceived previously. The difference of RNN and other deep learning models is that recurrent networks have two sources of input, the present and the recent past. With the combination of those two data types in the shape of a feedback loop, RNN can determine how to respond to new data. In order to handle more complicated structures with RNN, a generalized version of RNN has been proposed called a recursive neural network (again RNN) *(Socher et al., 2011)*. Recursive models have a deep hierarchical tree-like structure, rather than the chain-like arrangement of recurrent networks, and there is no time aspect to the input sequence.

There are countless hybrid deep learning approaches in the context of deep learning such that a combination of generative and discriminative techniques attempt to make a robust architecture to overcome difficulties of many complicated applications. To sum up this brief review of deep learning literature, we mention another network, which is a developed version of the Markov decision process and Monte Carlo method known as reinforcement learning (RL) *(Socher et al., 1998; Wang et al., 2016)*. The RL field involves many concepts such as agents, environments, states, actions, and rewards. In RL, an agent like a robot or a video game character has a current state, and it endeavors to take some

actions (the set of all possible moves for the agent in its environment) to change its state to a better one in order to get a reward depending on the action. In a feedback loop of state, action, and reward, the agent can learn to find the best solution for a problem to get the best reward (most accuracy). In other words, one RL model attempts to convert a compound probability distribution of rewards to a bunch of state-action pairs[1].

In order to explain our proposed methods for tongue contour and ultrasound probe tracking, we need to explain the theoretical framework of convolutional neural networks (CNNs) theoretical framework. As we mentioned earlier, CNNs are deep artificial neural networks that are used primarily for image analysis, such as classification, segmentation, and registration. Amongst these techniques, CNN deep learning models are used in many fields with significant performance in comparison with previous techniques. In the next section, the essential basics of CNNs are described briefly.

## 2.3 Image Segmentation using Deep Learning

### 2.3.1 Convolutional Neural Network

In many visual tasks, especially in biomedical image processing, the desired output is an image with segmented target regions, i.e., a class label is supposed to be assigned to each pixel. Nearly all the state-of-the-art image segmentation neural network structures are built upon convolutional neural networks (CNNs) to provide this kind of output image. The detailed information of how a CNN works and the building blocks of CNN is described in the following paragraphs.

**Convolutional operation:** It is a measurement of how much two functions overlap as one passes over the other. In the case of an image, the first function is a 2D matrix, which is the image, and the second function is a small 2D image (called the kernel). Kernel traverses on portions of the image to find the similarities between the kernel and the image and to extract specific features from the image depending on the shape of the kernel. Figure 2-7 shows applying one kernel $K$ to the image $I$. For each pixel in the output, values from element-wise multiplication of $I \times K$ are summed into one value, which is the output pixel. When convolution kernel is exerted on an image, the convolution kernel looks as though it is an observation window at the detection time, which gradually moves from the upper left corner of the image to the bottom right corner.

[1] There are many references for details of each model in the literature such as *(L. Deng, 2014; Géron, 2017; Goodfellow et al., 2016; W. Liu et al., 2017)*

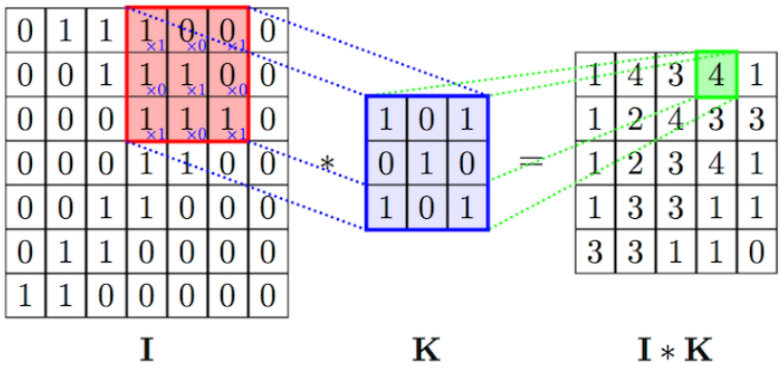



Figure 2-7. A sample of applying kernel on an image using convolutional operation. The output image is smaller than the input image. Values of $K$ are changed during the training of CNN to delineate valuable information from the input image *(Krizhevsky et al., 2012)*.

In CNN models, many kernels in several network layers, with different values and sizes, are applied to an image to extract features. As can be seen from Figure 2-7, the size of the input image is shrunk after each layer of convolution though the number of feature depth is increased, depends on the number of kernels. In general, a deeper CNN layer provides a smaller image with more depths (see Figure 2-8).

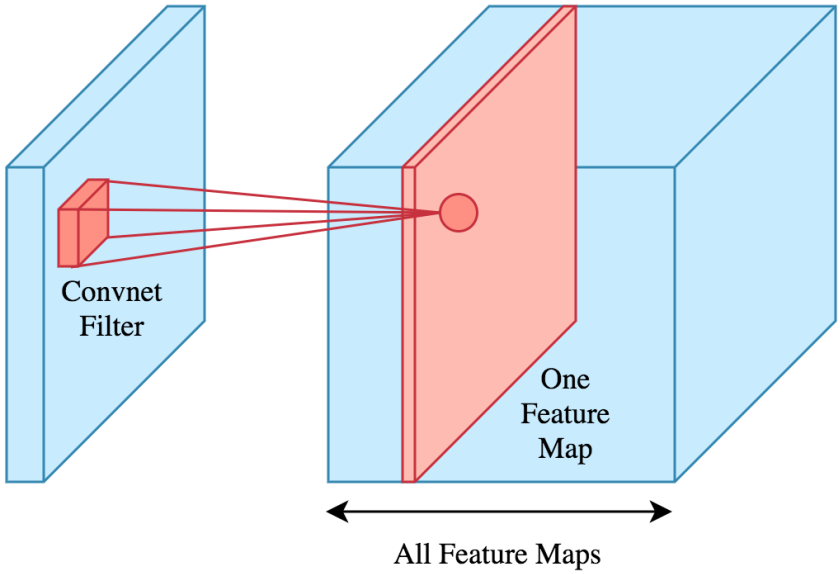



Figure 2-8. A typical convolutional layer. The left side is the input image with a small depth (e.g., an RGB image has a depth of three), and on the right is the output image with more depth and smaller image size *(Krizhevsky et al., 2012)*.

CNNs can be used to classify each pixel in an image individually, by presenting it with patches extracted around the particular pixel. A drawback of this naive *sliding-window*

approach is that the input patches from neighboring pixels have huge overlap, and the same convolutions are computed many times. Fortunately, the convolution and dot product are both linear operators, and thus inner products can be written as convolutions and vice versa. By rewriting the fully connected layers as convolutions, the CNN can take larger input images than it was trained on and produce a prediction map, rather than an output for a single pixel. The resulting *fully convolutional network* (FCN) can then be applied to an entire input image or volume efficiently.

**Pooling layer:** After obtaining the features of the image through the convolution layer, in theory, we can directly use these features to train a classifier, though it is a large computational challenge, and prone to the so-called over-fitting phenomenon. Over-fitting typically happens when the error on the training set is driven to a small value, but when new data is presented to the network, the error is significant. The network has memorized the training examples, but it has not yet learned to generalize new situations.

In order to further reduce the network training parameters and the over-fitting problem of the model, researchers use a down-sampling (pooling) convolution layer after each convolutional layer output. Pooling is usually in the following two ways (see Figure 2-9): 1) Max-Pooling: Select the maximum value in the pooling window as the sample value; or 2) Mean-Pooling: Sum all the values in the pooling window and take the average as the sample value.

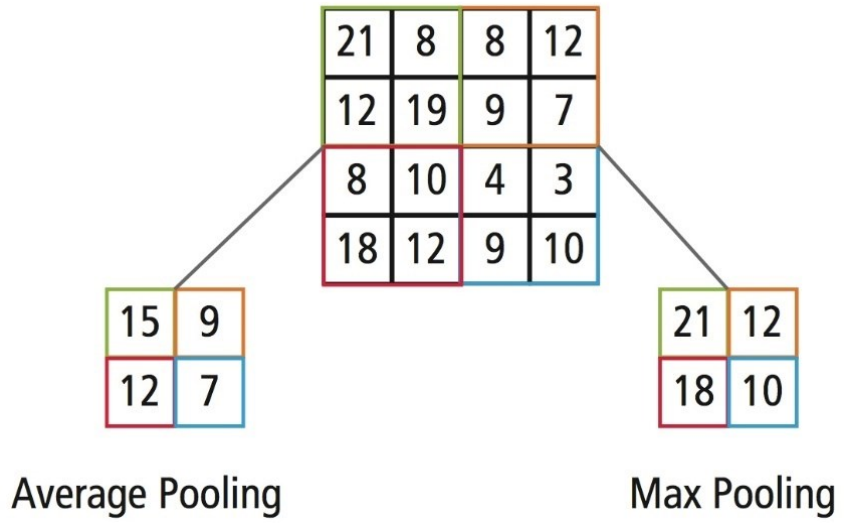



Figure 2-9. A sample of one pooling operation. Max pooling is more popular than average pooling due to its better performance in computer vision applications.

**Activation layer:** In order to add non-linearity to the output results, a function called the activation function is used for each layer of CNN. Non-linear factors are added to a

network because the expression of linear models is not sufficient to calculate complicated features. In other words, the model should simulate non-linear functions as is required for solving non-linear problems. The commonly used activation functions in deep learning context are sigmoid (Equation 2-3), hyperbolic tangent (Equation 2-4), and Rectified Linear Units (ReLU)(Equation 2-5). The graph of each activation function is illustrated in Figure 2-10.

$$f(x) = \frac{1}{1 + e^{-x}}$$ Equation 2-3

$$f(x) = tanh(x)$$ Equation 2-4

$$f(x) = max(0, x)$$ Equation 2-5

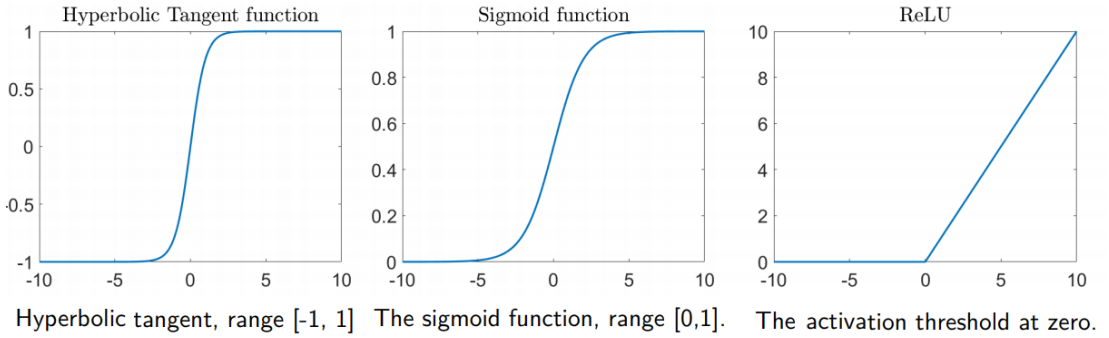



Figure 2-10. The illustration of some standard activation functions in deep learning.

**Fully connected layer:** The fully connected layer is equivalent to a classifier in the network, which calculates the class scores corresponding to output categories. In image segmentation, usually, a SoftMax function (Equation 2-6, where $k$ is the dimension of input $x$) is used to change network output values into probabilities.

$$f(x) = \frac{e^{x_i}}{\sum_{j=0}^{k} x_j}, i = 0,1,2, \dots, k$$ Equation 2-6

**Dropout layer:** The dropout layer was proposed by *(Srivastava et al., 2014)*. When training a neural network model, a dropout layer might be added between network layers in order to prevent the model from over-fitting. During training, the dropout layer can stop portions of the feature detectors from working, which can improve the generalization of the network. Figure 2-11 shows a neural network before and after employing dropout layers.

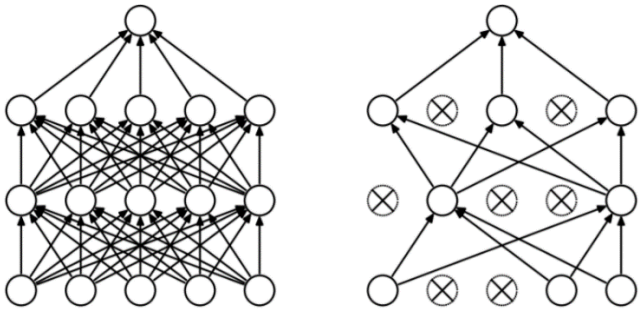


Figure 2-11. A neural network structure before and after applying dropout *(Srivastava et al., 2014)*. Non-connected circles on the right image are deactivated nodes (neurons).

**Over-fitting:** This is a problem that arises during the training of a neural network. When a model is over-fitted to the training data, it loses its capability of generalization. The model has learned the training data, including noise, to such a great extent that it has failed to capture underlying general information. CNNs have a large number of weights to be trained, and therefore overfitting can occur due to training too few training examples. Using dropout layers, a neural network might be trained with a lower over-fitting problem (see Figure 2-11).

In the dropout approach, nodes and their connections are randomly dropped from the network. Dropout constrains the network adaptation to the training set. Consequently, it prevents the network weights from too much fitted on training data. Therefore, the difference in performance between training and testing decreases[1] considerably.

**Backpropagation (BP):** Training a neural network is usually accomplished in two phases: *forward pass* and *backward pass*. In the first phase, an image is fed into a network, and the first network layer outputs an activation map. Then, this activation map is the input to the first hidden layer, which computes another activation map. Again another activation map is computed using the values of this activation map as inputs to the second hidden layer. Carrying out this process for every layer will eventually yield the network output. In the backward pass, network weights are updated using the backpropagation approach.

The CNN requires to adjust and update its kernel parameters, or weights, for the given training data. Backpropagation is an efficient method for computing gradients required to perform gradient-based optimization of the weights in neural networks using

[1] Note that dropout layers are used during training only, not during validation or testing

gradient chain rule *(Srivastava et al., 2014)*. In this method, for each section of the network, one gradient is calculated, and by multiplications of each section, the gradient of the whole network can be found. A predefined loss function $L$ is used to minimize the difference between the input and desired output. The goal is to adjust the weights so that the loss function value decreases. This is achieved by calculating the derivative with respect to the weights of the loss function.

For instance, in supervised image segmentation, training a neural network is executed by finding the difference between labeled and input data using a loss function, which is defined as an error. Then, this error is backpropagated from the output layer toward the input, and an optimization method adjusts weights in the network layers. In this way, the entire error of the network is minimized iteratively. Therefore, in general, there are two computational phases for the training of a neural network, the forward pass and the backward pass in which the weights are updated to find the optimum value of the loss function.

**Optimization method:** As mentioned before, for updating the weights of a neural network, loss function error should be minimized. Optimization techniques, iteratively, find the optimum values of a function by changing the values of its variables with different strategies *(Mohammad Hamed Mozaffari et al., 2016)*. Several optimization algorithms are designed for training neural networks such as Stochastic Gradient Descent (SGD) *(Ruder, 2016)* and Adam optimization algorithm *(Kingma & Ba, 2014)* (refer to *(Goodfellow et al., 2016)* for more details about optimization algorithms).

**Loss function:** The difference between a training data, after it has propagated through the forward network, and a labeled data (desired output) is calculated by a loss function, as a criterion of algorithm performance. Two widely used loss functions for deep learning applications are Mean Square Error (MSE) (Equation 2-7) and Cross-Entropy (Equation 2-8), where $x_i$ is the *i-th* neuron output and $\hat{x}_i$ is the *i-th* desired output.

$$L = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2 \qquad \text{Equation 2-7}$$

$$L = \frac{1}{N}\sum_{i=1}^{N}(\hat{x}_i \times \ln(x_i) + (1 - \hat{x}_i) \times \ln(1 - x_i)) \qquad \text{Equation 2-8}$$

**Batches:** When a deep learning model is training, applying the whole dataset can cause a big issue in terms of computational resources such as memory and processing units. For this reason, instead of training a network by feeding the whole dataset for each iteration and computing parameters, errors, and gradient for the whole dataset, it is trained by

batches of data randomly selected from the dataset. Each time that all data samples are given through batches, it is called an epoch[1]. For this kind of training, there is a specific kind of optimization method which minimizes the average of errors from all batches named Mini-batch optimization, such as mini-batch SGD.

### 2.3.2 Dilated Convolution

The consecutive combination of convolutional and pooling layers in encoding part of DCNN model results in a significantly lower spatial resolution of output feature maps, typically by a factor of 32 which is not the desired resolution for the semantic segmentation purposes for modern deep learning architectures *(L.-C. Chen et al., 2017)*. Several ideas have been proposed to reconstruct an input-size segmented output from the course feature map of the encoding part in a DCNN. Interpolation (up-sampling) could be the first solution, whereas each pixel of the feature map is repeated to increase the image size *(Long et al., 2015)*. This method of down-sampling and up-sampling with inevitable losing information is not beneficial for the task of segmentation when boundary delineation with a high resolution is required.

Transposed convolution (sometimes called deconvolution) has been introduced to solve this issue and to recover the low-resolution prediction maps *(Zeiler et al., 2010)*. Transpose convolutional layer operates opposite of a convolution layer where each pixel of the feature map is expanded to the kernel size and superimposed with its neighbor. Although deconvolution improved segmentation results *(Noh et al., 2015)*, it still suffers from the checkerboard problem *(Odena et al., 2016)* as well as increasing the number of learnable parameters. Other techniques such as indexed un-pooling *(Badrinarayanan et al., 2015; Zeiler & Fergus, 2014)* require memory for saving positions while max-pooling is applied on a previous feature map.

Using dilated convolution (sometimes called atrous convolution) while the dilation factor is increased monotonously through layers, it revealed that the receptive field could be effectively expanded with keeping a spatial resolution. However, the sparsity of dilated kernels does not always cause performance improvement, especially for small objects and details *(Hamaguchi et al., 2018)*. To solve this problem, one solution is to decrease the dilation factor throughout the decoding path of the network model, similar to the number of kernels in the U-NET model *(Hamaguchi et al., 2018)*. Furthermore, the use of dilated

[1] For an example of using batch, imagine a dataset with 200 images, batch size 20, epoch 5. In this way, the model is applied on 10 batches each of which contains 20 images (200 / 20 = 10 batches). So, after network training, it sees the whole images for 5 times, it means (5 × 200 = 1000) 1000 images.

convolutions introduces gridding artifacts that are similar to the checkerboard artifacts. In *(Yu et al., 2017)*, several ideas have been proposed to decrease the gridding problem, including increasing and decreasing dilation factors in the forward path of a DCNN. Following these techniques, we proposed two DCNN models (named BowNet) that benefited from standard and dilated convolutions in one network model.

Figure 2-12 illustrates samples of decoding strategies using interpolation, deconvolution, and dilated convolution. Dilated convolution has been successfully utilized in recent studies with the improvement of segmentation accuracy in comparison with standard architectures *(L.-C. Chen et al., 2017, 2018; Hamaguchi et al., 2018)*.

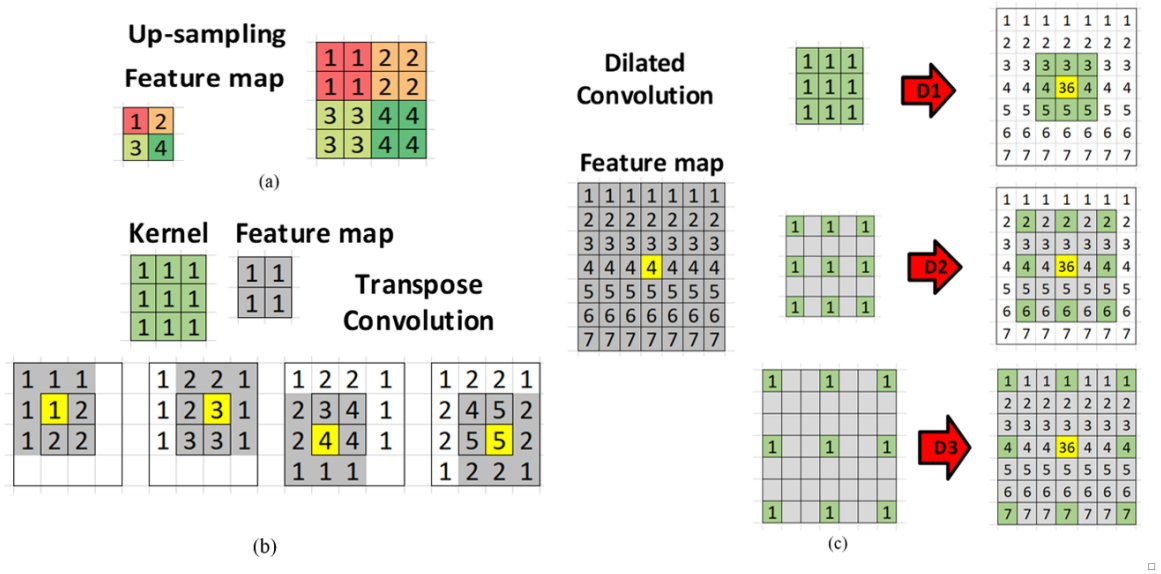



Figure 2-12. Several ideas for improving the low-resolution feature map to acquire an output image with the input-size image. (a) and (b): Up-sampling using linear interpolation and transpose convolution, respectively. (c) Dilated convolution with different dilation factor of 1, 2, and 3.

### 2.3.3 Semantic Segmentation Methods

There are numerous tasks in computer vision, including image classification (assigning a label, e.g., image of "cat" or "dog"), object localization (indicating a bounding box for every object in an image), and detection (appointing both label and bounding box for each object simultaneously). In a higher level of these tasks, semantic image segmentation, also known as scene parsing, the problem is to assign a semantic label for every pixel in an image. As a taxonomy, scene parsing techniques are divided into semantic or instance segmentation. Aim of semantic segmentation is to assign one label

for multiple objects of the same type, while, in instance segmentation, the goal is to select distinct labels for individual objects in an image *(L.-C. Chen et al., 2016)*.

Semantic segmentation is a fundamental step in a large group of applications, from scene understanding in self-driving vehicles *(Z. Wu et al., 2019)* to the delineation of lesions in medical image analysis *(Jiang et al., 2018)*. The main complication of semantic segmentation is closely related to scene and label variety *(H. Zhao et al., 2017)* as well as the requirement of laborious works for manual labeling. However, in recent years, several groundbreaking deep learning methods based on Fully Convolutional Networks (FCNs) *(Long et al., 2015)* have been exploited for the problem of semantic segmentation with astonishing advancements in several benchmarks *(L.-C. Chen et al., 2018; He et al., 2017; X. Liu et al., 2019)* over systems relying on hand-crafted features *(L.-C. Chen et al., 2018)*.

The crucial elements for success of semantic segmentation methods are one of the two factors *(L.-C. Chen et al., 2016; H. Zhao et al., 2017)* of using multi-scale features, where features concatenated from intermediate layers using skip connections (e.g., spatial pyramid pooling) *(H. Zhao et al., 2017)* or utilizing multi-scale input images to a shared network *(L.-C. Chen et al., 2016; G. Lin et al., 2017)*. Recently, combinations of encoder-decoder architectures *(Ronneberger et al., 2015)* and other techniques such as spatial pyramid pooling *(H. Zhao et al., 2017)*, architectures with dilated convolutions *(L.-C. Chen et al., 2017, 2018; M. Hamed Mozaffari et al., 2019)*, and post-processing methods *(L.-C. C. Chen et al., 2017)* provide sharper object boundaries for several image segmentation benchmarks.

For the last decade, Deep Convolutional Neural Network (DCNN) *(Krizhevsky et al., 2012)* has been the method of choice for semantic segmentation tasks in many fields of science *(Garcia-Garcia et al., 2017; Guo et al., 2016; Litjens et al., 2017; Mamoshina et al., 2016; Zhou et al., 2017; X. X. Zhu et al., 2017)*. The groundbreaking innovation in semantic segmentation happens after substituting the fully connected layer by a fully convolutional layer in popular recognition networks such as LeNet and AlexNet *(Long et al., 2015)*. In this way, dense image classification methods could be adopted for semantic segmentation of images with arbitrary sizes and with a lower computational cost (succeeding architectures called as FCN based models). As a general taxonomy, novel FCN based image segmentation methods (semantic, instance, or panoptic *(Kirillov et al., 2019)*) can be divided into two complementary approaches depend on their network architecture: I) Encoder-Decoders and II) Pyramid structures. These two methods aim to exploit contextual information as well as providing precisely delineated target regions.

**Encoder-Decoders:** inspired by auto-encoders *(Zhai et al., 2019)*, this kind of model consists of one encoder-decoder pair as the core of their network. The encoder blocks encode information into feature space *(Chaurasia & Culurciello, 2017)*, where the spatial dimension of feature maps is progressively decreased where deeper layers capture a more extended range of knowledge. On the other hand, the decoder aims to retrieve the detailed categorical semantic information with a detailed spatial dimension. For instance, a decoder can be a simple bilinear up-sampling *(Long et al., 2015)* or using deconvolution *(Noh et al., 2015)* along with skip connections to use encoder results in the decoding process *(Badrinarayanan et al., 2015; Ghiasi & Fowlkes, 2016; Ronneberger et al., 2015; J. Zhao et al., n.d.)*. Recently, dilated (also called Atrous) *(L.-C. Chen et al., 2014)* convolution is used prevalently in semantic segmentation literature to reduce the number of parameters at the same time with a better receptive field *(L.-C. C. Chen et al., 2017)*.
**Pyramid structures:** CNN has a poor performance in dealing with scale variance, whereas the segmentation of a large object, but not complicated, in an image is difficult for CNN models *(Y. Xu et al., 2014)* (see Figure 6-23 for an example). On the other hand, different studies demonstrate the importance of using multi-scale information for having a high-quality segmentation result *(L.-C. Chen et al., 2016, 2018)*. Pyramid structure models (usually on top of an encoder-decoder) aim to exploit multi-scale features *(L.-C. Chen et al., 2018; Takikawa et al., 2019; H. Zhao et al., 2017)* and image *(L.-C. Chen et al., 2016; G. Lin et al., 2017)* information to acquire sharper object boundaries as a scale invariance architecture. For instance, *(L.-C. Chen et al., 2018; H. Zhao et al., 2017)* proposed pyramid pooling modules to use the multi-scale feature while *(L.-C. Chen et al., 2016; G. Lin et al., 2017)* employed multi-scale inputs with significant improvements in well-known benchmarks.

On top of previous techniques, pre-processing and post-processing approaches provide even more detailed segmentation results in numerous studies. In the former one, the region of interest is elected by techniques such as region proposals in *(He et al., 2017)*, cropping in *(L.-C. Chen et al., 2016; Ronneberger et al., 2015)*, and attention modules in *(L.-C. Chen et al., 2016)*, while in the later one post-processing is used to improve results such as using CRF in *(L.-C. C. Chen et al., 2017; L.-C. Chen et al., 2014)* or overlap-tile strategy in *(Ronneberger et al., 2015)*, to name a few. In many studies, the attention of researchers is on performance speed *(Nekrasov et al., 2018; Paszke et al., 2016)* without compromising the accuracy.

Embedding different Convolutional neural networks (CNNs) in cascade (deeper *(L.-C. Chen et al., 2017; Fu et al., 2019; Jégou et al., 2017)*) and cascode (shallower *(L.-*

*C. Chen et al., 2018; M. Hamed Mozaffari et al., 2019; H. Zhao et al., 2017)*) configurations as well as utilizing multi-scale feature maps and input images advances the performance of CNN models for semantic segmentation tasks. Furthermore, most of the existing techniques in this literature benefit from one encoder-decoder architecture as their central network component *(Badrinarayanan et al., 2015; Chaurasia & Culurciello, 2017; Noh et al., 2015; J. Zhao et al., n.d.)*, which demonstrates considerable improvement in both accuracy *(L.-C. Chen et al., 2018; He et al., 2017; H. Zhao et al., 2017)* and speed *(Nekrasov et al., 2018; Paszke et al., 2016; Siam et al., 2018)*. However, in designing almost all state of the art semantic segmentation models, the default routine is to adopt a publicly available classification encoder model (see Table 5-2), trained on a large dataset such as ImageNet *(J. Deng et al., 2009)*. Therefore, the impact of elaborating one personal model, pre-trained on the current task as a relevant feature extractor, is always ignored in many studies *(Siam et al., 2018)*.

In our experimental studies, we demonstrate that the performance of different scenes parsing frameworks strongly depends on their pre-trained encoder block despite their successful results in many studies. As a result of this dependency, there is not yet one prevailing deep learning model applicable to different types of datasets with satisfying results. Towards designing a general deep learning model for semantic segmentation datasets, we propose a new convolutional module inspired by human peripheral vision *(Rosenholtz, 2016)* (named RetinaConv). To use the module, we employ RetinaConv in an encoder-decoder architecture (called IrisNet).

## 2.4 Deep Learning in Medical Image Segmentation

Advances of technology (since medical data such as images could scan and load into a computer) help researchers to build systems for automated analysis. The first trace of supervised techniques using training data in medical science can be found at the end of the 1990s *(Litjens et al., 2017)*. A crucial step in designing many automatic systems in medicine is the extraction of discriminant features from the images, which is still done by human researchers. Thus, deep learning algorithms aim to let computers learn the features and omit that manual work.

Depending on the application, images might contain valuable information, and accurately extracting this information is usually a challenging task. Image segmentation is an essential preparatory process in almost any image processing technique when one image is delineated into desired target objects. As an optimum segmentation, pixels that

contribute to each class should be similar in terms of information *(M.H. Mozaffari & Lee, 2017; Mohammad Hamed Mozaffari & Lee, 2016)*. For example, input data such as medical images are given to a CNN as an end-to-end model, and it provides the probability of presence or absence of disease for each pixel of the image, and the consequent outcome is a segmented image of disease, healthy, and background regions.

Automatic segmentation of anatomical structures in ultrasound imagery can significantly help researchers to extract valuable and accurate information from each ultrasound image. Automatic ultrasound image segmentation *(Hamed Mozaffari et al., 2019; Mohammad Hamed Mozaffari & Lee, 2016)* has been a challenging topic for recent years, and the advent of deep learning techniques is a promising solution for this challenging task. Up to now, the research on deep learning techniques has attracted a great deal of attention, and it shows that deep learning algorithms, particularly the development of unique CNN-based segmentation architectures *(K. Xu et al., 2017)*, are powerful enough for solving many problems in pattern recognition and data mining such as image segmentation.

Since 1990, machine learning approaches have been becoming popular in medical image analysis, such as active shape models and the use of statistical classifiers for feature extraction *(Litjens et al., 2017)*. Although many breakthroughs happened in the deep learning field recently, there has been a gradual shift in medical image analysis toward using deep learning techniques. Deep learning methods such as CNN were employed first for medical image studies in research by *(Lo et al., 1995)*. However, deep learning techniques for medical image analysis have been investigated thoroughly only during the last recent years, after many achievements in many computer visions tasks. For decades, image processing approaches have been dominant solutions for applications such as ultrasound image segmentation *(Alison & Boukerroui, 2006)*. Although image processing techniques could achieve successful outcomes in various research studies, they suffer from many difficulties, such as being specific to one particular application, being slow in performance, usually need manual initialization and human intervention, and are not suitable for real-time tasks.

In the field of medical image segmentation using deep learning, the most popular network model is U-NET, published by *(Ronneberger et al., 2015)*. Other deep learning techniques have also been exploited in medical image segmentation successfully *(Lai, 2015; Litjens et al., 2017; Shen et al., 2017)* such as RNNs *(Xie et al., 2016)*, LSTM *(Stollenga et al., 2015)*, and GANs *(Xue et al., 2018)*. Figure 2-13 shows several famous architectures

in medical imaging. In the following section, some well-known examples of those architectures for image segmentation applications are described in more detail.

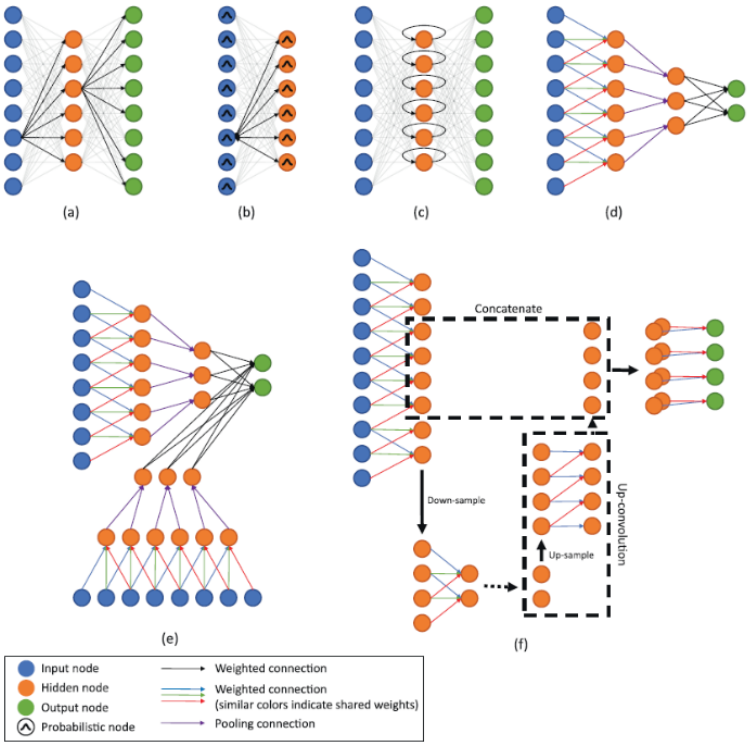



Figure 2-13. Node graphs of deep learning architectures commonly used in medical imaging. (a) Auto-encoder, (b) Restricted Boltzmann machine, (c) Recurrent neural network, (d) Convolutional neural network, (e) Multi-stream convolutional neural network, (f) U-NET (with a single down-sampling stage) *(Litjens et al., 2017)*.

**Fully convolutional networks (FCNs):** FCNs are advanced architectures for many computer visions tasks, specifically for image segmentation problems. In the first attempt to improve segmentation output results, *(Long et al., 2015)* designed, trained, and introduced the FCN model for pixel-wise semantic segmentation. In order to make a network suitable for pixel-wise segmentation, specific layers have been modified, which enabled the generation of segmented output maps. The idea of using a modified CNN for pixel-wise segmentation is not new. *(Matan et al., 1992)* modified the LeNet network for the purpose of recognizing strings of digits. The goal of FCN is to produce *segmented images* with the same size as the original input image. FCN labels each pixel of the input image with one of $C$ colors, where $C$ is the number of classes, we are segmenting.

The basic idea behind a fully convolutional network is that all of its layers are convolutional layers, including the last layer. In the output layer, the number of channels

is equal to the number of classes. For example, if we are classifying each pixel as one of fifteen different classes, then the final output layer has a dimension of height × width × 15. Using a SoftMax loss function in the last layer of an FCN model, the probability that each pixel belongs to one class is calculated in the output image. The first application of FCN in medical image processing was with *(Moeskops et al., 2016)*, who trained a single FCN to segment brain MRI. They reported that an accurate segmentation result was obtained in sample images. Figure 2-14 shows the original architecture of the FCN network for image segmentation of dog, cat, and background.

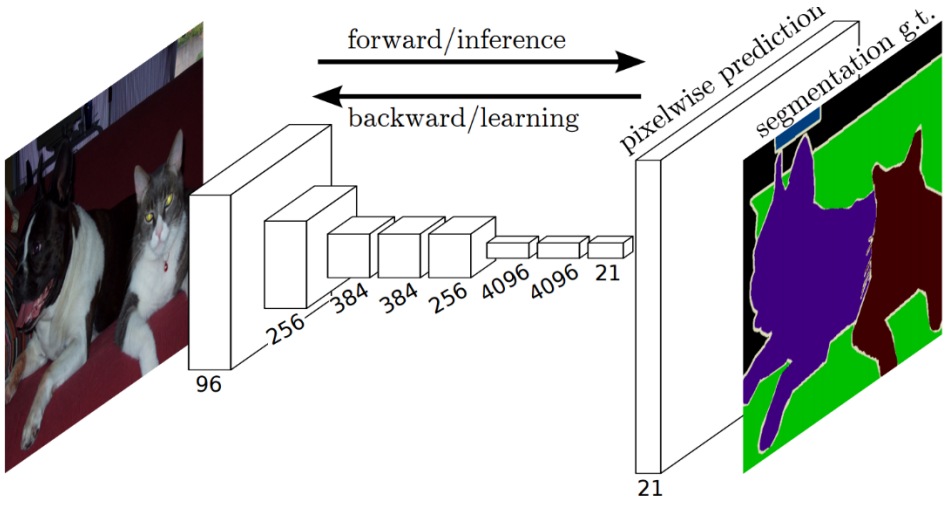



Figure 2-14. The architecture of the FCN network. A fully connected layer in the last section of the network is not efficient for image segmentation results *(Long et al., 2015)*. Instead, a fully convolutional layer can address the resolution problem of the output image.

Recently, FCN models are used in medical image segmentation applications but with two drawbacks. First, it is quite slow because the network must be run separately for each patch, and there is a lot of redundancy due to overlapping patches. Secondly, there is a trade-off between localization accuracy and the use of context. Larger patches require more max-pooling layers that reduce the localization accuracy, while small patches allow the network to see a little context. For improving the performance and bring out the potential of CNN models, many complex architectures had been proposed, such as U-NET and SegNet, for medical image analysis tasks.

**U-NET:** In medical image analysis, U-NET, which is a novel CNN architecture, has become the most well-known technique for image segmentation task *(Ronneberger et al., 2015)*. This architecture is a unique version of several other segmentation models *(Badrinarayanan et al., 2015; Long et al., 2015; Noh et al., 2015)* such that a new operator,

the so-called skip connections between opposing convolution and deconvolution layers, is added to the encoding-decoding network. This operator concatenates outputs from the contracting and expanding paths resulting in a higher resolution output images.

U-NET works with very few training images, and it yields more precise segmented images compared to FCN and previous methods. The main idea is to supplement a usual contracting network by successive layers, where up-sampling operators reconstruct output image from predictions of the encoder block. In order to alleviate the loss of information by pooling layers, high-resolution features from the encoding path are combined with the successive decoding layers. This process is done with the concatenation layer, which combines two layers with the same dimension by randomly chosen sampling. Figure 2-15 shows the complete layout of the U-NET structure *(Ronneberger et al., 2015)*, such that each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower-left edge of the box. White boxes represent copied feature maps.

In an extended paper by *(Falk et al., 2019)*, it is shown that a full 3D segmentation can be achieved by feeding U-NET with a few 2D annotated slices from the same volume. Other authors have also built derivatives of the U-NET architecture. It is worth mentioning that although U-NET is one of the most cited papers with satisfying accuracy, the large structure of this model is a limit in many cases, especially when real-time performance is put into consideration.

**V-net:** *(Milletari et al., 2016)* proposed a 3D-variant of U-NET architecture, called V-net, performing 3D image segmentation using 3D convolutional layers. In their implementation, at the end of each stage (similar to the U-NET structure), the appropriate strides decrease the resolution. The encoding path is included in the left part of the network, while the right part decodes the signal until reaching the original size. As Figure 2-16 shows, the left part of the network is separated into several stages in which different resolutions are put into operation. Each stage comprises one to three convolutional layers *(Milletari et al., 2016)*. The most original feature for the V-net is that they formulate each stage such that it learns a residual function: the input of each stage is (a) used in the convolutional layers and processed through the non-linearities and (b) added to the output of the last convolutional layer of that stage in order to enable learning a residual function.

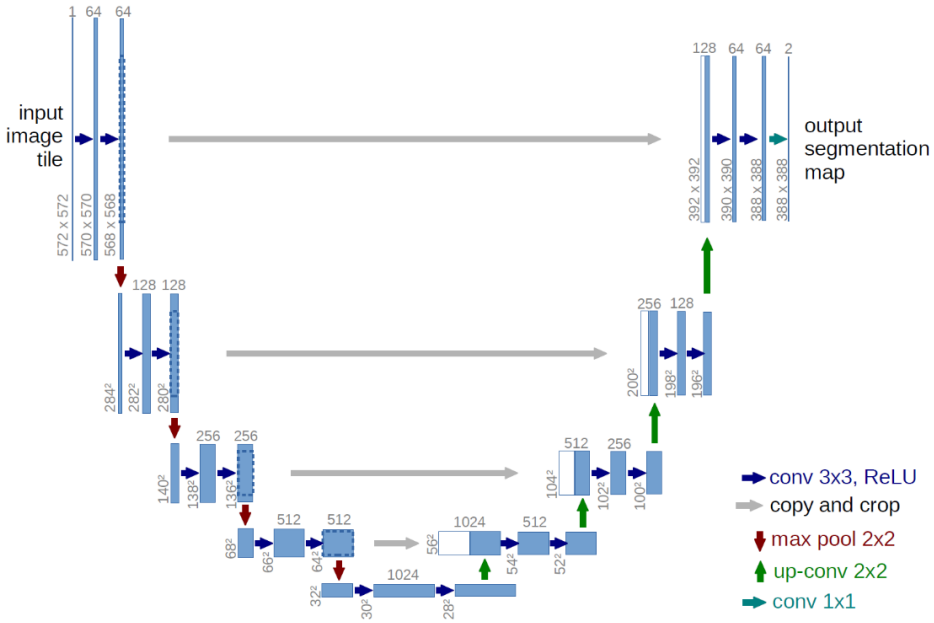



Figure 2-15. The architecture of the U-NET network. Portions of encoding layers are conveyed for decoding purpose that increases the output accuracy *(Ronneberger et al., 2015)*.

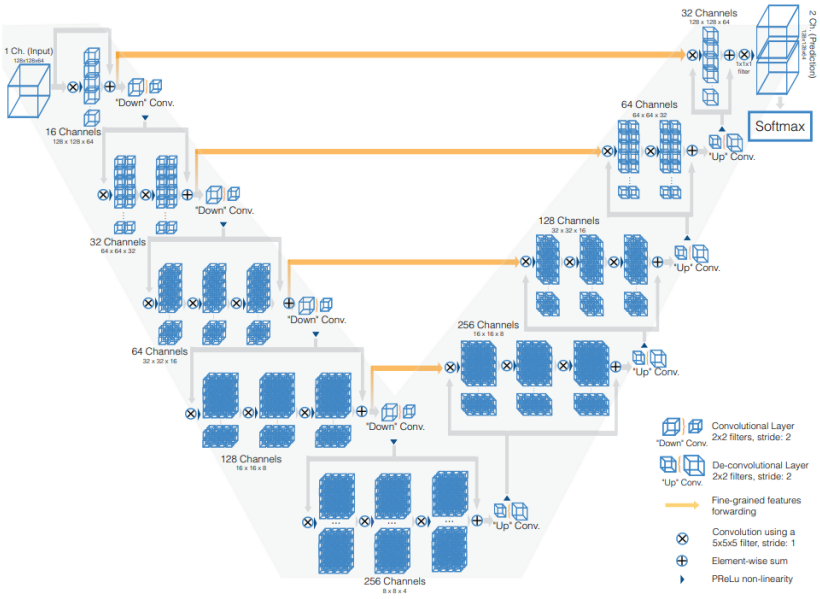



Figure 2-16. The architecture of the V-Net network for 3D image segmentation *(Milletari et al., 2016)*.

**SegNet:** Another innovative deep learning architecture after U-NET for image segmentation is SegNet. In this architecture (see Figure 2-17), three consecutive convolutional layers are used to have a better receptive field (wider). Instead of the concatenation of down-sampling images with the up-sampling section, like U-NET, SegNet conveys max-pooling indices from the encoding part to the decoding. Having indices of

pooling layers will help to reconstruct the output image with more detailed information about the location of each pixel before applying the pooling layer. Consequently, the output image would be less blurred and with sharper details.

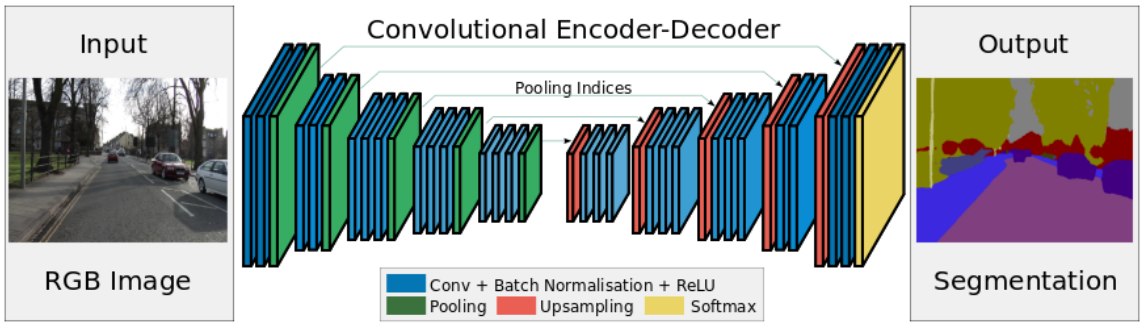



Figure 2-17. The architecture of the SegNet network. Pooling indices are kept for the reconstruction step in decoding layers *(Badrinarayanan et al., 2015)*.

**DeepLab:** DeepLab architectures have been proposed for semantic segmentation task to delineate sharper segmented regions from an input image. There are several variants of DeepLab in the literature (v1 *(L.-C. C. Chen et al., 2017)*, v2 *(L.-C. C. Chen et al., 2017)*, v3 *(Szegedy et al., 2016)*, v3+ *(L.-C. Chen et al., 2018)*) with successful results for many important semantic segmentation benchmarks. The main idea of DeepLab models is to use dilated convolution in order to decrease trainable parameters in the model as well as providing a more efficient receptive field for the model. In the first three versions of the DeepLab model, the focus is on the effective usage of dilated convolutions for semantic segmentation applications. In the most advanced version of DeepLab V3+, information is extracted from the input image in different paths similar to the BowNet models, which we proposed in our methodology. DeepLab models are powerful for semantic segmentation problems, but the usage of DeepLab models for the medical image segmentation has not been fully exploited yet.

## 2.5 Object Tracking using Deep Learning

Object tracking, as its name indicates, is the problem of estimating the trajectory of an object in the image plane as it moves around a scene. It is one of the topics within the field of machine vision whose purpose is to track one or more objects in video frames. Object tracking is a challenging task due to abrupt object motion, changing appearance patterns of both the object and the scene, non-rigid object structures, object-to-object and object-to-scene occlusions, and camera motion *(Yilmaz et al., 2006a)*.

The plethora of powerful computers, developments of algorithms and training techniques, and advances of sensor technology in video cameras, which made it available and inexpensive, has stirred a great deal of attention in object tracking methods. As *(Yilmaz et al., 2006a)* have mentioned, there are three main steps for video analysis: detection of an object or the region of interest, tracking of the object in each frame of the video, and assessment of tracking result to understand the behavior of the object. Accordingly, object tracking methods might be used for tasks such as human identification, detection of suspicious activities from surveillance systems, interaction with computer using gesture or eye gaze in video data, monitoring of traffic flow and gathering of real-time statistics, and autonomous vehicle driving which needs video path planning and obstacle avoidance capabilities.

As another definition, object tracking can be considered as assigning consistent labels to the tracking objects in different frames of a video *(Yilmaz et al., 2006a)*. A tracker can provide object-based information, such as orientation, area, position, or shape. Difficulty of object tracking is due to several factors like loss of information from projection of 3D to 2D image, noise in images, complex motion of objects, non-rigid or articulated nature of objects, partial and full object occlusions, complex object shape, illumination change in scene, and real-time processing requirements *(Yilmaz et al., 2006a)*.

From our investigation in recent object tracking studies, we can classify object tracking methods into two major groups as a new taxonomy: image processing techniques and machine learning techniques. In the former one, the goal is to track one object by imposing constraints on the motion and appearance of the object. In this strategy, all tracking algorithms assume that the object motion is with no abrupt changes and smooth with constant velocity or acceleration based on prior information *(Yilmaz et al., 2006a)*. The primary purpose of machine learning techniques is to designing effective decision models or extracting robust features *(Qi et al., 2016)*. For instance, in recent studies, CNNs have been successfully utilized for object recognition, detection, and tracking *(Qi et al., 2016)*. In the following subsections, we explain some famous techniques in each category. A successful tracker should handle different scales of the target object, illumination changes, background clutter, partial occlusions, and operate in real-time *(Kalal et al., 2012)*.

### 2.5.1 Image Processing Techniques

In this tracking scenario, objects should be defined first as anything that is of interest for further analysis, such as the face of humans in video data, vehicles in traffic

images, pedestrians in one street, products in a factory, or stars in the sky. The shape and appearance of objects can represent their characteristics, and they are commonly employed for tracking in image processing techniques. Representation of an object by its shape can be defined by points (centroid or multiple points), rectangular or circular patch, elliptical or part-based multiple patches, the skeleton of the object, complete object contour along with control points, and silhouette which is the region inside the object contour. Other techniques represent features of an object as its appearance instead of shapes such as probability density, templates, active appearance models, and multi-view appearance models *(Yilmaz et al., 2006a)*.

The selection of the right feature is an essential part of tracking, and it should be unique so that the object can be easily recognized in the feature space. Standard features for image processing are color, edges, optical flow, and texture, which are defined priorly depend on each application. Although feature selection used to be done manually, there are many attempts to design automatic feature selection methods. Object tracking is highly related to the object detection mechanism either in every frame or when the object first is seen in the video. In this way, an object is detected in one frame, and that information is used for other frames, or temporal information between different frames are used to detect an object.

There are many object detectors in the literature. Point detectors are popular techniques try to find interest points that are constant in terms of feature characteristics in all frames such as Harris, Kanade-Lucas-Tomasi (KLT), and Scale Invariant Feature Transform (SIFT). Background subtraction and image segmentation are two other well-known approaches for object detection using image processing approaches. In general, for tracking of an object using image processing techniques, first, the object should be detected using its unique features, then it is tracked employing one representation technique such as point, primitive geometric, or contour model. Therefore, object detection and recognition are of great importance for object tracking.

For example, researchers have proposed human face tracking techniques with acceptable results. In a famous study by *(Vinet & Zhedanov, 2010; Viola & Jones, 2004)*, a variation of intensity on human face between eyes in the horizontal and vertical direction was utilized as a feature to detect a face in each frame. They could propose a fast and robust face tracking method that is capable of real-time performance with higher accuracy than similar techniques. A comprehensive survey of object tracking methods can be found in a study by *(X. Li et al., 2013; Smeulders et al., 2014; Yilmaz et al., 2006a)*.

### 2.5.2 Machine Learning Methods

As we mentioned in a previous section, image detection first should be done to detect an object in one image, and then it can be tracked. Machine learning idea can make this process of detection easier by providing a generalized model which is trained in advance for the current application. In this approach, the goal is to train a model (for example, a neural network) by different views of an object as training data. Then, the trained model is used to track that object in new images. Numerous machine learning models have been used for object tracking so far *(Avidan, 2004; Babenko et al., 2011; Schroff et al., 2015; F. Tang et al., 2007)*. Recent successful results of convolutional models in many machine vision applications triggered researchers to use deep architectures for object tracking. Employing deep learning techniques for tracking purposes has been reported in many studies *(B et al., 2016; K. Chen & Tao, 2018; Denil et al., 2012; Gan et al., 2015; Kahou et al., 2017; Qi et al., 2016; Zhai et al., 2019)*.

Object tracking can be addressed using similarity learning such that a similarity function learns to compare training frames along with labeled object trajectories to a candidate image of the same size. Then, in the testing stage, the position of the object in a new image can be found by searching for the maximum similarity value between the past appearance of the object and different position candidates in the new image. Therefore, in video data, initial position of the object is recognized in the first frame and is considered as the exemplar.

Famous example of this approach is *Siamese* architecture *(Bertinetto et al., 2016; Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, 1994; Koch & Koch, 2015; Taigman et al., 2014; Tao et al., 2016; Zagoruyko & Komodakis, 2015)* which applies an identical transformation to both input images and then combines their representation using a similarity metric to calculate the differences. Results of deep learning challenges such as VOT2015 *(Fern et al., 2015)* for the problem of object tracking revealed that convolutional networks are able to track objects in video data accurately and rapidly if suitable training data could be provided for them. New ideas such as one-shot training have been proposed to address this need for training data, and this topic is still an open area for research *(Caelles et al., 2017)*.

# Chapter 3 Ultrasound technology in Linguistics

Ultra-high frequency sound, both emitted and received by piezoelectric crystals of ultrasound transducer/probe creates echo patterns that are decoded as an ultrasound image. Ultrasound signal penetrates and traverses linearly through materials with uniform density but reflects from dense substances such as bone. With the ultrasound transducer held under the chin and with the crystal array lying in the mid-sagittal plane of the head, the ultrasound screen displays information about the superior surface of the tongue from the tongue root to near the tip along the mid-sagittal plane *(F. Campbell et al., 2010)* (see Figure 3-1).

One particular linguistically valuable property of ultrasound imaging is the capability of simultaneous visualization of the front and back of the tongue *(Wilson et al., 2006)*. For instance, in the production of some consonant such as /l/ and /r/, the time intervals between gestures of different tongue's regions (e.g., the tongue tip, tongue dorsum, and tongue root) are essential, and they depend on the position of the consonant in the syllable (i.e., onset versus coda) *(F. Campbell et al., 2010; F. M. Campbell, 2004; Tateishi & Winters, 2013)*. Therefore, an L2 learner can use ultrasound imaging to understand the source of those mistiming errors, results in more accurate pronunciation.

The efficacy of using ultrasound imaging on the pronunciation of North American /r/ and /l/ phonemes has been proven by depicting the complexity of the tongue's shape for L2 language learners *(Adler-Bock et al., 2007; Wilson et al., 2006)*. Although ultrasound has been utilized successfully in L2 pronunciation training, interpretation of raw ultrasound data for language learners, especially in a real-time video stream, is a challenging task. Moreover, to understand ultrasound videos accurately, a language learner ought to know the tongue's gestures and structures as well as the interpretation of ultrasound data *(Bliss et al., 2016)*.

For phonetic and speech training, a mid-sagittal view of ultrasound imaging is widely adopted, as it displays relative backness, height, and the slope of various regions of the tongue *(M. B. Bernhardt et al., 2008)*. As mentioned before, localization and interpretation of tongue gestures from B-mode images is a real challenging task due to low signal to noise (SNR) ratio, high speckle noise corruption, and ultrasound artifacts *(L. et al., 2012)*. Tongue movements are fast with rapid changes during the speech process, and because of the limits of acquisition frame rate, tongue contour is not visible in all frames during the recording *(Stone, 2005)*. Besides, manual labeling is needed for at least

initialization in many research studies *(Laporte & Ménard, 2018)* even during the test stage of the system.

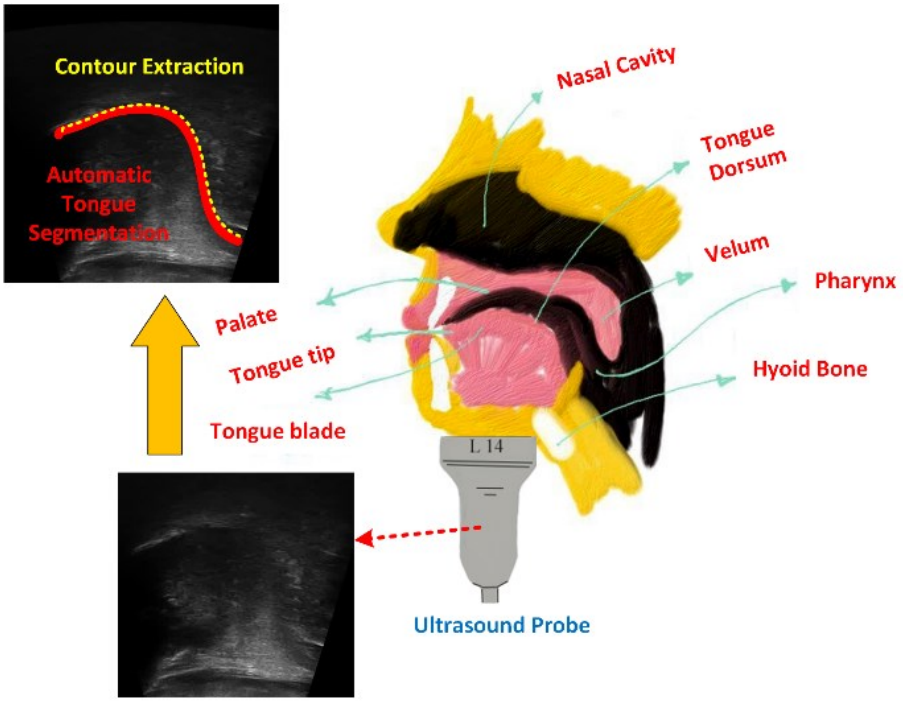



Figure 3-1. Anatomy of the human tongue in mid-sagittal view. Tongue surface is relatively a sharp transient from dark to bright in ultrasound images. Interpretation of Ultrasound video frames by non-expert L2 learners requires knowledge of tongue anatomy and understanding ultrasound data. The sample ultrasound video frame is cropped into squared size (which side is the tip of the tongue?). The tongue contour can be highlighted (delineated) for a better understanding of the tongue gestures.

## 3.1 Ultrasound Tongue Contour Tracking

When at rest, the tongue displays an unremarkable gross morphology, but dynamically it is a highly mobile, deformable, and precise organ, routinely deploying, in extremely rapid succession, diverse, subtly different movements with great combinatorial flexibility. Visualization of tongue gestures with these various dynamics states is the first pace in the treatment of patients with speech sound disorders (SSD) *(Preston et al., 2017)*. At the same time, the kinetics of the tongue during speech is complicated to see directly or to visualize using conventional cameras or optical motion trackers, since other oral structures completely obscure its functional movements.

Medical ultrasound systems have favorable real-time capabilities, are fast and relatively inexpensive, portable, and non-invasive. Ultrasound imaging has been utilized

in many studies to visualize at a reasonably rapid frame rate (more than 60 Hz). For tongue movement investigation, it enables researchers to study and capture the subtle and swift movement of the tongue during speech production. In order to quantify and analyze the tongue shape deformations using ultrasound data, the tongue surface must be delineated from image sequences with a curved contour, which is usually defined under the brightest and longest continuous region in each video frame.

Illustrating the tongue model or at least a curve on top of the tongue dorsum can significantly help the understanding of the tongue structure in ultrasound data, and it alleviates the interpretation problem. Furthermore, automatic tongue contour tracking opens a way for further research toward 3D visualization of tongue instead of only 2D images in real-time. Moreover, it allows researchers to compare their methods quantitatively in terms of accuracy. Accordingly, in order to quantify and analyze the tongue shape deformations using ultrasound data automatically, the first step is to delineated tongue surface from image sequences *(Wrench et al., 2011; Zharkova, 2013)* as a curved contour which is usually defined under the brightest and longest continues region in each video frame *(Lee et al., 2015)*.

Hence, having a thoroughly automatic method with robust and accurate results for long duration image sequences is the goal of many studies in this field, and any successful result would be beneficial for further studies. Utilizing previous tracking software packages, such as EdgeTrak, extracting tongue contour of a single frame can take over seconds, which is not suitable for real-time applications like language training purposes. For instance, overlaying videos of the speaker's head and ultrasound data have revealed its benefits for a better understanding of tongue displacements for language learners *(Abel et al., 2015)*. Ultrasound images are not clear enough, and depicting the whole tongue for students can confuse them, especially when one student aims to compare his/her tongue with the tongue of a native speaker, which they are not the same in terms of shape and size. In the following, the studies which have been done for the tongue contour tracking are reviewed, the methodologies including the traditional models which use active contours, statistical approaches, a physical model of the tongue, and the state of the art machine learning approaches.

**Active Contour Models:** Active contour model or snake *(Kass et al., 1988)* is the first proposed method that gained success in tackling the highly deformable structures of human organs such as the tongue surface. The locations of a discrete set of vertices, defining the deformable contour, are estimated in a way that the contour optimizes two energy functions, expressing the expected appearance of the image in the vicinity of the

contour (via an external energy functional) as well as constraints on the allowed deformations (via an internal energy functional). Figure 3-2 shows a sample of an active contour model that is going toward the tongue dorsum iteratively. Vertices (Control points) are updated using image gradient values of other points in the neighbor.

More specifically, a snake is an energy minimizing problem, deformable spline influenced by image constraints and forces that pull it towards object contours and internal forces that resist deformation. Snakes may be understood as a special case of the general technique of matching a deformable model to an image employing energy minimization *(Kass et al., 1988)*. In 2D, a snake is a curve which can be represented as: $C(s) = (X(s), Y(s))$ where $S \epsilon [0, 1]$. The curve moves through the image range to minimize a specified energy function. In traditional snakes, the energy is usually formed by internal forces and external forces are described in Equation 3-1.

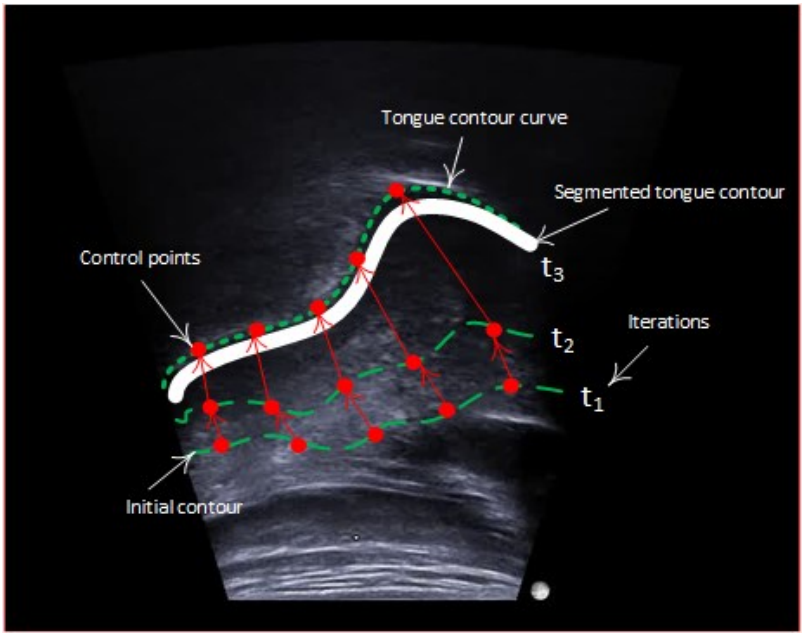



Figure 3-2. Sample of an iterative process in the active contour technique. Users should initialize at least one tongue contour first by selecting several control points near the tongue dorsum region, and one curve is interpolated between those points.

$$E_{snake} = E_{internal} + E_{external} \qquad \text{Equation 3-1}$$

$E_{internal}$ tends to elastically hold the curve together (elasticity forces) and to keep it from bending too much (bending forces). This energy is defined in Equation 3-2 where $C_s$ and $C_{ss}$ represent the first and second derivative of the image pixels in different directions,

respectively. The tension and rigidity of the snake can be controlled by the coefficients $\alpha$ and $\beta$.

$$E_{internal} = \frac{1}{2}\int_s^{\alpha} |C_s|^2\, ds + \int_s^{\beta} |C_{ss}|^2\, ds \qquad \text{Equation 3-2}$$

$E_{external}$ intends to pull or push the curve towards the edges. Typically, the external forces consist of potential forces. This energy is defined where $E_{image}$ represents the negative gradient of a potential function, and it is generally the image force where $I$ denotes the image and $X = X(x,y) = [x,y]_t$.

$$E_{external} = \int_s E_{image}\big(C(s)\big)ds \qquad \text{Equation 3-3}$$

$$E_{image}(X) = -|\nabla I(X)|^2 \qquad \text{Equation 3-4}$$

The differential equation can be solved using variational calculus and the Euler-Lagrange. Then, the solution to this force balance represents the final position of the snake. The differences in the way the energy function is established will result in different snakes.

$$\alpha C_s - \beta C_{ss} - \nabla E_{image} = 0 \qquad \text{Equation 3-5}$$

Although traditional snakes have found many applications, they are intrinsically weak in four main aspects.

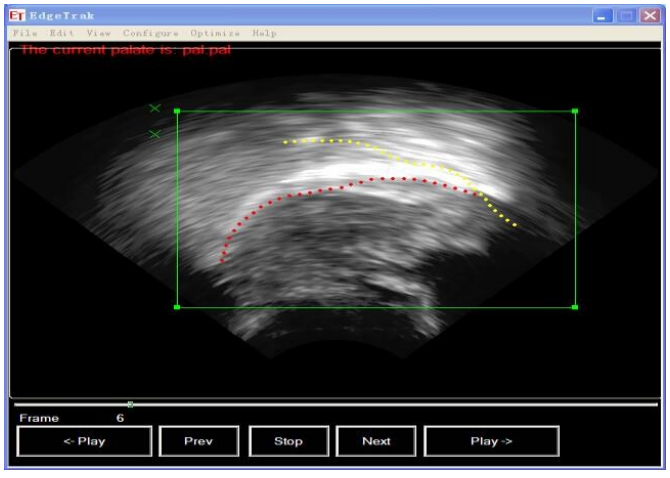



Figure 3-3. EdgeTrak software for ultrasound tongue tracking, the method of the EdgeTrak is the active contour model, which is used for helping annotation in this research project *(M. Li et al., 2005)*.

- They are sensitive to parameters.
- They have a small capture range.
- They have difficulties in progressing onto boundary concavities.
- The convergence of the algorithm is mostly dependent on the initial position.

*(Akgul et al., 1999)* used another snake formulation for tongue segmentation and tracking in ultrasound video sequences, and they proposed the use of dynamic programming for contour optimization in each frame. The same research group used a modified version of external energy to include a band energy factor that constrains the tongue surface to lie immediately below a bright white band, thereby reducing the likelihood of the snake latching onto speckle noise in the ultrasound image. This method is publicly available as the EdgeTrak software *(M. Li et al., 2005)*. EdgeTrak requires the user to identify a few points near the tongue contour in the first frame of the sequence and then iteratively proceeds to optimize the snake for that frame and copy it to be used as an initial guess for the next frame, thereby enforcing some degree of temporal consistency.

EdgeTrak is widely adopted for ultrasound image enhancement, and it has gained popularity in annotating the tongue ultrasound images. However, it fails in the presence of rapid tongue curvature increases, and it occasionally produces tongue shapes that are uncharacteristic of those encountered during speech. One way to overcome this problem is to enforce global shape constraints on the segmented tongue contour by fitting an active shape or active appearance model *(Cootes et al., 2001)* of the tongue to the ultrasound data. As it proposed by *(Roussos et al., 2009)*, an active appearance model was built using a training database of segmented X-ray and ultrasound images to respectively establish shape constraints and characterize ultrasound image texture in the vicinity of the tongue. Active shape models (ASMs) can also be used to constrain and iteratively drive snake optimization *(Roussos et al., 2009)*.

**Statistical Model-based Methods:** Strong temporal consistency constraints for tongue tracking were introduced by *(L. et al., 2012)*. A statistical model-based method was proposed in their work, and a high-order Markov random field optimization was used for tongue contour extraction. Their method was implemented in their publicly available software TongueTrack, and they modeled the tongue image segmentation as a global optimization problem over a higher order Markov random field where vertices represent the tongue contour evolving in *2D+t*, and its edges represent adjacency constraints in space (along the tongue contour) and in time. Recently, in a similar way, advanced

recurrent deep learning techniques such as LSTM has been utilized similarly for the problem of tongue contour tracking *(Aslan & Akgul, 2019)*.

The tongue contour *nodes* are connected via *edges* within and across ultrasound frames. The tracking boils down to labeling each node with a displacement vector moving the tongue contour to its optimal position in each frame. The model is mathematically elegant, and it accounts for spatial (i.e., shape) and temporal (i.e., motion) constraints in a flexible manner. It also allows future frames to condition past ones, which may be useful to resolve some ambiguities. This method does not require any training data. As a result, it incorporates very little contextual knowledge. This method is computationally intensive, and it needs powerful computational and memory units. Figure 3-4 shows simulated samples of TongueTrack software, which uses a high-order Markov random field optimization method.

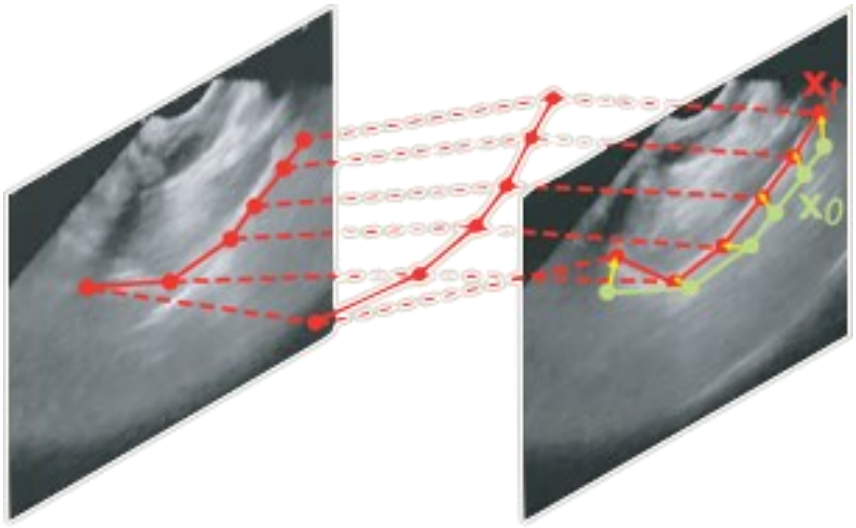



Figure 3-4. Sample images from TongueTrack software, which uses a high-order Markov random field optimization method *(L. et al., 2012)*.

**Physical Model-based Methods:** Instead of relying on heuristics or segmented datasets to constrain tongue shape and kinematics, *(Loosvelt et al., 2014)* used a biomechanical model as the prior information for tongue tracking in ultrasound data. Rather than tracking a contour in the image, their method involves fitting the biomechanical model to a set of tracked point features belonging to the entire tongue (not necessarily the surface). Their preliminary results suggest that realistic tongue motion constraints improve tongue tracking when parts of the tongue are invisible in the image.

The proposed biomechanical models consist of computing 3D object deformations using physical laws. In order to have an accurate result, the continuum mechanics

formulation (conservation laws and matter continuity) are chosen for the model. Various numerical techniques exist to solve this set of equations. The finite element method (FEM) is used to compute the mechanical equations of the tongue model. In this method, the discrete geometry for dividing the complex problem into small elements is defined firstly. Then the problem becomes a mechanical system expressed in ordinary differential equations, followed by the establishment of the boundary conditions that the system must satisfy. One of the system equations is the constitutive law that depends on material properties. Finally, the solution of the system is presented as a fitted model on the ultrasound tongue image *(Loosvelt et al., 2014)*.

The physical model technique relies on tracking point-based landmarks in ultrasound images, where it is a notoriously error-prone task. The manual initialization also is an essential part of this approach. Further validation is needed to assess the performance of this kind of method (see Figure 3-5 for a physical model of the tongue for ultrasound data).

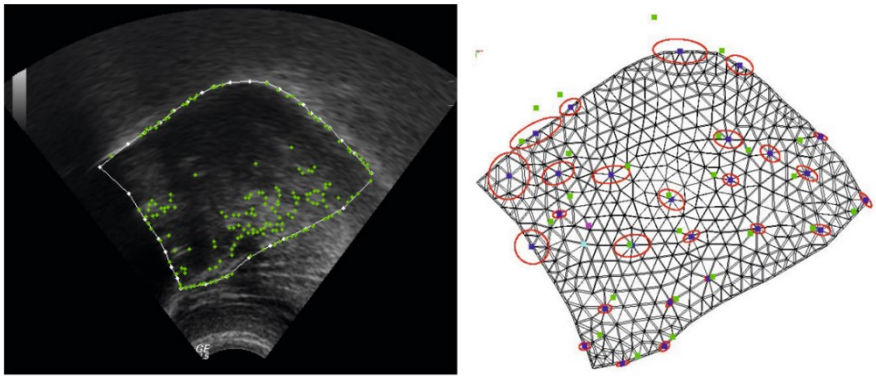


Figure 3-5. The physical model fits on the ultrasound tongue image. Positions of nodes (red dots) are updated to find the best position for fitting the tongue model on the image *(Loosvelt et al., 2014)*.

**Machine Learning-based Techniques:** If a dataset of segmented ultrasound images is available, as is assumed by the shape-constrained snake methods described before, then this dataset can also be used to learn the relationship between image features and the tongue contours using machine learning methods. *(Fasel & Berry, 2010)* investigated this approach using deep neural networks, leading to the publicly available Autotrace software. In Autotrace, one deep neural network is trained on individual segmented ultrasound images to build an abstract and compact generative model of the relationship between image features and the location of contour vertices. Another deep neural network

establishes the relationship between unlabeled image data and this abstract model so that labels (i.e., tongue segmentations) can then be inferred based on image data only.

*(Fabre et al., 2015)* proposed a similar approach, based on a simpler neural network architecture, that establishes a relationship between the principal component representation of images and that of the target tongue contour. These methods are entirely based on machine learning, and they require a few manual initializations. They also perform segmentation independently on each image, thereby avoiding the propagation of segmentation errors from one frame to the next, however, neglecting the temporal consistency of tongue shape as a useful source of regularizing information.

The model of deep neural networks proposed in *(Geoffrey E. Hinton et al., 2006)* is based on the stacking of Restricted Boltzmann Machines (RBMs). Again, as mentioned in the previous chapter, Restricted Boltzmann Machine is a neural network composed of a layer with visible units and a layer with hidden units connected through directional links (weights), which are symmetric. The output probability of a hidden unit depends on the weighted activations of the units in the visible layer (and vice-versa, since the connections are symmetric). In this method, an autoencoder (a system that takes an input vector $x$ and maps it to a hidden representation $h$ using an encoder network) is used, followed by a decoder network (which reconstructs the input from the hidden representation). A common way of constructing an autoencoder is through RBM. An RBM is a specific type of Markov random field in which the $V$ dimensional input vector $x$ and $H$ dimensional hidden feature vector $h$ are modeled.

So far, a variety of techniques have been tested for tongue contour tracking such as active contour models *(Akgul et al., 1998, 1999; Ghrenassia et al., 2014; Iskarous, 2005; Laporte & Ménard, 2015; K. Xu, Yang, et al., 2016)*, graph-based technique *(L. Tang & Hamarneh, 2010)*, and machine learning-based methods *(Fabre et al., 2015; L. et al., 2012; L. Tang & Hamarneh, 2010)*. However, deep learning is used for tongue contour tracking only with employing generative deep neural network models, and there is no evidence of using discriminative techniques such as CNN for tongue tracking. In a study by *(K. Xu et al., 2017)*, CNN has been utilized just for tongue gesture target classification. In similar studies *(J. Berry & Fasel, 2011; J. J. Berry & James, 2010; Fasel & Berry, 2010; Jaumard-Hakoun, Xu, Leboullenger, et al., 2016; Jaumard-Hakoun, Xu, & others., 2015)*, tongue contour was extracted automatically using deep belief networks (DBNs) and deep auto-encoder *(Jaumard-Hakoun, Xu, Leboullenger, et al., 2016)* which are generative models. In this thesis, we comprehensively investigate the usage of CNN architectures for tongue tracking. A complete review of tongue contour tracking approaches have been performed

by *(Laporte & Ménard, 2018)*, and a comparative study of many previous methods can be found in an investigation by *(K. Xu, Gábor Csapó, et al., 2016)*.

## 3.2 Artificial Intelligence for Second Language Acquisition

Communication skill is one of the essential aspects of the second language (L2) acquisition so that it is often the first indication of a language learner's linguistic abilities *(Bird et al., 2018)*. Pronunciation directly affects many social interaction skills of a speaker, such as communicative competence, performance, and self-confidence. Previous studies revealed that other aspects of L2 learning, such as word learning, can be developed easier by accurate pronunciation *(Johnson et al., 2018)*. However, pronunciation learning is one of the most challenging skills to master for adult learners *(Abel et al., 2015)* in traditional classroom settings. There is often no explicit pronunciation instruction for language learners because of limited time and lack of knowledge of effective pronunciation teaching and learning methods *(Abel et al., 2015)*.

Standard practice for a language learner, outside of the class, is to imitate a native speaker's utterances in front of a mirror limited to lip and jaw movements along with hearing of recorded acoustic data. In practice, it is difficult for an L2 learner to utter new words correctly without any visual feedback of a native speaker and lack of awareness of how sounds are being articulated *(Abel et al., 2015)*, especially in cases where the target sounds are not easily visible (Bird et al., 2018). The positions and movements of the tongue, especially all but the most anterior part, cannot be seen in the traditional approach of listening and repeating word's pronunciations *(Bliss, Burton, et al., 2017)*. Learners can only have proprioceptive feedback of their tongue location depends on practicing sounds (vowels, liquids, or others) and the amount of contact their tongue makes with the teeth, gums, and palate *(Wilson et al., 2006)*.

Visual feedback approaches have been developed over the past decades to enable L2 learners to see moving speech articulators during a speech or a training session, benefiting from a range of tools called Electronic Visual Feedback (EVF) *(Wilson et al., 2006)* including ultrasound imaging, electromagnetic articulography (EMA), and electropalatography (EPG) *(Bliss, Abel, et al., 2018)*. Among those technologies, ultrasound imaging is particularly non-invasive, safe, offer high dimensional continuous real-time data with acceptable framerate, portable, versatile, user-friendly, widely available, and increasingly affordable. Furthermore, ultrasound technology is capable of recording and illustrating the whole regions of the tongue (although the mandible

sometimes obscures the tongue tip *(Bird et al., 2018)*) during both dynamic and static movements. Other imaging modalities such as MRI and X-ray (more specifically cinefluorography) are also capable of showing a mid-sagittal view of the tongue. However, these techniques are often prohibitively expensive, non-accessible, and invasive *(Wilson et al., 2006)*.

Deep learning-based methods and their applications in image processing literature, such as object detection and image segmentation, have been a research hotspot in recent years. Deep learning methods are powerful in automatic learning of a new task, while unlike traditional image processing methods, they are capable of dealing with many challenges such as object occlusion, transformation variant, and background artifacts *(L.-C. C. Chen et al., 2017; Guo et al., 2016; Z.-Q. Zhao et al., 2018)*. Recent technology-assisted language learning methods, such as multimodal approaches using ultrasound imaging, have been successfully employed for language pronunciation teaching and training, providing visual feedback of learner's whole tongue movements and gestures *(Abel et al., 2015; Antolik et al., 2019; B. Bernhardt et al., 2005; Bird et al., 2018; Gick et al., 2008; Hueber, 2013; Yamane et al., 2015)*. However, this technology is still far from commercializing for use in all language training institutes. The authors observe several main gaps in the current literature:

- Ultrasound-enhanced multimodal methods require manual pre-processing and post-processing works, including image enhancement, freezing for further analysis, and superimposing of ultrasound frames and side-face profile video frames. Moreover, manual synchronizing is required between ultrasound frames, video frames, and audio data. All manual works are time-consuming, subjective, and error-prone tasks, which require a knowledge of video and audio editing toolboxes *(Abel et al., 2015; Bird et al., 2018)*. The performance quality of each study is evaluated only by qualitative investigation, and the quantitative study of the method is only limited to post-processing stages, while it requires freezing target frames following cumbersome manual works.
- It is a challenging task for a non-expert language learner to interpret ultrasound data in real-time *(Bliss et al., 2016)* without having any knowledge of tongue structure and ultrasound imaging. Coloring the tongue regions in ultrasound data *(Bird et al., 2018)* cannot be an efficient and generalized approach for this problem due to the different ultrasound image data characteristics as well as the requirements of additional manual work.

- Our experimental study revealed that language learners could understand the gestures of their tongue better in real-time using an ultrasound-enhanced multimodal visualization approach than previously recorded offline systems. Instead of using a guideline on the screen (usually on the palate *(B. Bernhardt et al., 2005)*), using an automatic tracking technique, a language learner can perceive the real-time location of the tongue respect to landmarks of the face.
- Due to the dependency of previous ultrasound multimodal biofeedback methods on the subject's face specifications *(Hamed Mozaffari et al., 2019)*, those studies are not applicable for other medical applications such as ultrasound augmented reality, and manual works for primary system's modifications are inescapable.
- Previous stabilization methods have been designed only for one specific type of ultrasound probe without any flexibility for a user's head position. To overlay ultrasound video frames on the user's face, the user's head should be stabled during recording, which makes the training session non-conformable. Furthermore, different ultrasound helmets and probe stabilizers are not accessible in any research and teaching departments.

As a taxonomy, application of ultrasound imaging technology in linguistics can be classified into the following categories: I) tongue contour extraction and tracking *(Laporte & Ménard, 2018)*, II) visualization of tongue gestures for language pronunciation teaching and learning *(Abel et al., 2015; Bird et al., 2018)*, III) silent speech interfaces *(Denby et al., 2010)*, and IV) speech disorder and hearing impaired diagnosis and rehabilitation *(B. Bernhardt et al., 2005; Davidson, 2006)*. The main goal of our proposed ultrasound-enhanced multimodal visualization system, benefiting from deep learning approaches for object tracking and tongue segmentation, is to alleviate the difficulties of previous studies.

# Chapter 4 Datasets

The first step for supervised training of a deep learning model is to prepare a dataset. Annotated tongue ultrasound datasets are rare, and researchers annotate each dataset depend on their application. In this project, we created several annotated datasets for semantic segmentation and probe tracking as well as used popular benchmarks from semantic segmentation literature. In this chapter, we report the characteristics of each dataset used in this project. Details of each dataset, including data augmentation, are explained in the next chapters.

## 4.1 Ultrasound Tongue Image Sequences

Few well-known ultrasound tongue datasets are publicly available (see Table 4-1 for a list of datasets). As can be seen from the table, there is no available dataset with annotated ground truth images for tongue contour tracking. In this project, we made two datasets where each image is annotated using our customized annotation software. Two experts (Authors of the thesis and Mr. Shuangyue Wen who worked on the same topic) put few points near the tongue surface, following the B-spline method, which fits a curve on these points *(M. Hamed Mozaffari et al., 2019)*. In order to decrease the effect of human error in annotation process, results of our annotation from each expert were monitored several times frame by frame.

Table 4-1. List of a few accessible publicly available ultrasound tongue datasets. International Phonetic Association (IPA) chart is a complete list of all phonemes in the English language.

| Name of dataset | Institute | Data type | Annotation |
|---|---|---|---|
| SeeingSpeech *(Lawson, E., Stuart-Smith, 2019)* | University of Glasgow | IPA/Video | N/A |
| Sigma *(Cai et al., 2011)* | ESPCI Paris | Image | N/A |
| eNunciate *("enunciate UBC," 2019)* | University of British Columbia | IPA/Video | N/A |
| Ultrasound in Speech Training *(Adler-Bock et al., 2007)* | University of British Columbia | Video | N/A |
| UltraSpeech *(Fabre et al., 2017)* | GIPSA Lab, France | Video | N/A |
| OttawaSpeech *(M. Hamed Mozaffari et al., 2019)* | University of Ottawa | Video | Yes |

For the problem of automatic tracking of tongue contours in ultrasound video data, our created datasets are made by different distributions. The reason is to have two different datasets for the generalization ability assessment of deep learning networks. The first dataset, from the University of Ottawa (OttawaSpeech), contains several videos of two English native speakers *(M. Hamed Mozaffari et al., 2019)*, and the other from the SeeingSpeech project dataset *(Lawson, E., Stuart-Smith, 2019)*, comprises of several downloaded videos of two English native speakers. In general, there are many videos in SeeingSpeech database from the two speakers. Due to the time-consuming procedure of data annotation, we utilized only several randomly selected videos from that database. Figure 4-1 presents two sample images from these two datasets after cropping and annotation. The annotation process took several days using our semi-automatic annotating software. Note that data was cropped in several sizes of 64×64, 128×128, 256×256, 512×512 and enhanced by omitting non-relevant information in data, such as palate reference curve in SeeingSpeech videos.

In order to have two standard datasets with different characteristics, we used the method of *(X. Y. Liu et al., 2006)* (Informed under-sampling). In this technique, random images are selected from the whole dataset. Then, the average image of each dataset is calculated using averaging of all image's intensities. The distance between each sample and the average sample provides a distance score. Figure 4-2 and Figure 4-3 illustrate the process of using this method for having two datasets with maximum variations. From Figure 4-4, it can be seen that SeeingSpeech is a more homogeneous dataset than the OttawaSpeech dataset with more heterogeneous images before the undersampling process. These comparisons is in terms of different data distribution of samples from each database's average image. Note that we balanced the two databases before dividing them into training and testing datasets. The final results of the informed under-sampling procedure are two similar databases but with some sort of variation in distribution for testing generalization ability of our models.

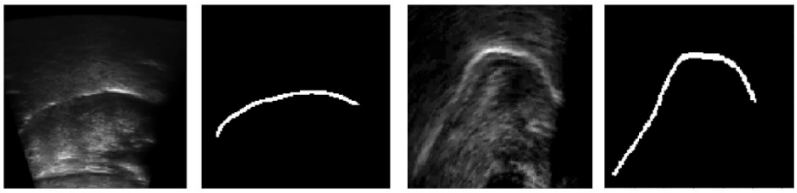


Figure 4-1. Sample images accompany with their corresponding ground truth labels from OttawaSpeech (two left images) *(M. Hamed Mozaffari et al., 2019)* and SeeingSpeech datasets (two right images) *(Lawson, E., Stuart-Smith, 2019)*.

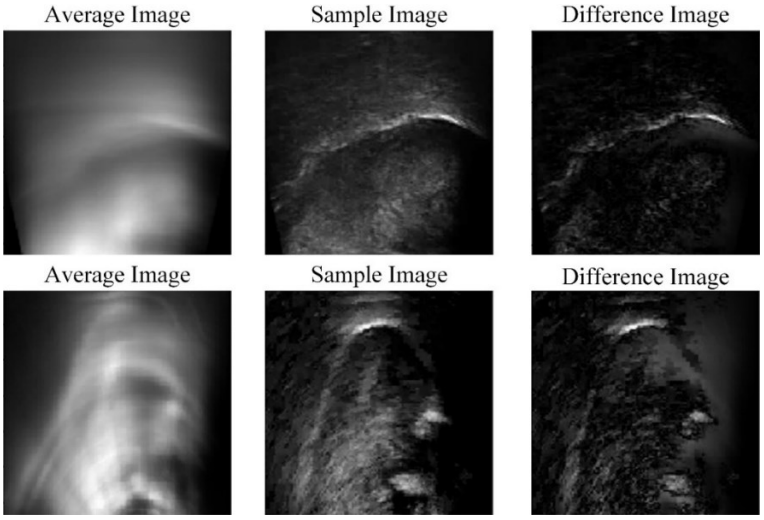



Figure 4-2. Informed under-sampling procedure *(X. Y. Liu et al., 2006)* to enhance datasets in terms of variation. Before undersampling: Left column: the average image from OttawaSpeech and SeeingSpeech datasets, respectively. Middle column: one sample image from each dataset. The right column: the difference between the sample image and the average one.

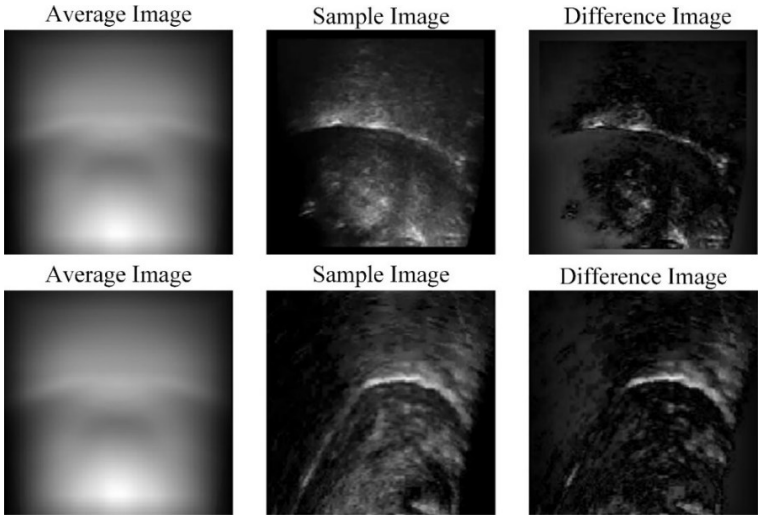



Figure 4-3. Informed under-sampling procedure *(X. Y. Liu et al., 2006)* to enhance datasets in terms of variation. After undersampling: Left column: the average image from OttawaSpeech and SeeingSpeech

datasets, respectively. Middle column: one sample image from each dataset. The right column: the difference between the sample image and the average one.

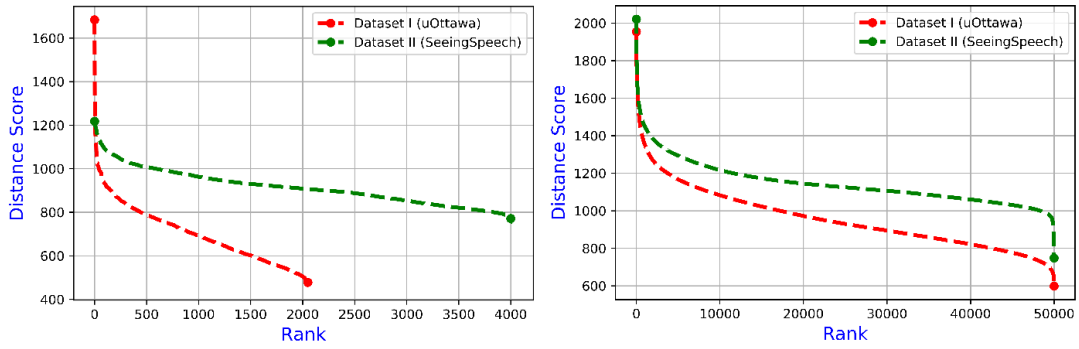



Figure 4-4. Left image: Ranking OttawaSpeech and SeeingSpeech datasets in terms of similarity distance score calculated the distance from each image to the average image. Right image: the same ranking values for datasets after informed under-sampling and data augmentation.

We selected 2050 images from each dataset after applying the informed under-sampling, 2000 images with the highest-ranking score, and 50 images from the smallest score *(M. Hamed Mozaffari et al., 2019)* to have a variety of data distribution for our databases. Table 4-2 shows the number of images in each dataset after offline augmentation and informed undersampling. It is noteworthy to mention that we use these datasets in different investigations with different parameter values such as different augmentation methods (on-line, during network training and off-line, prior to network training stage) in the next chapters. Table 4-2 presents a sample dataset configuration, ratios, and augmentation method. Note that test set is only used for testing with no augmentation, and it is completely fresh for the trained network model.

Table 4-2. Datasets information after informed undersampling and offline augmentation.

| | Randomly selected | Informed undersampling | Offline augmentation | Training Validation (%90/%5) | Testing (5%) |
|---|---|---|---|---|---|
| OttawaSpeech | 2058 | 2050 | 50,000 | 45,000/2500 | 2500 |
| SeeingSpeech | 4016 | 2050 | 50,000 | 45,000/2500 | 2500 |

It is noteworthy to mention that annotated masks are curved with the optimum diameters (i.e. tongue surface region in ultrasound data). For the first time in this field, we annotated curves with different diameters, such as using the whole tongue region instead of tongue contour in our preliminary experiments. Results were not significantly better for bigger masks, while the manual annotation process is laborious for accurate

annotation of the whole tongue. For this reason, we selected the 5 pixels diameter curves for our experiments. Figure 4-5 illustrates different annotation masks for tongue segmentation using different diameters.

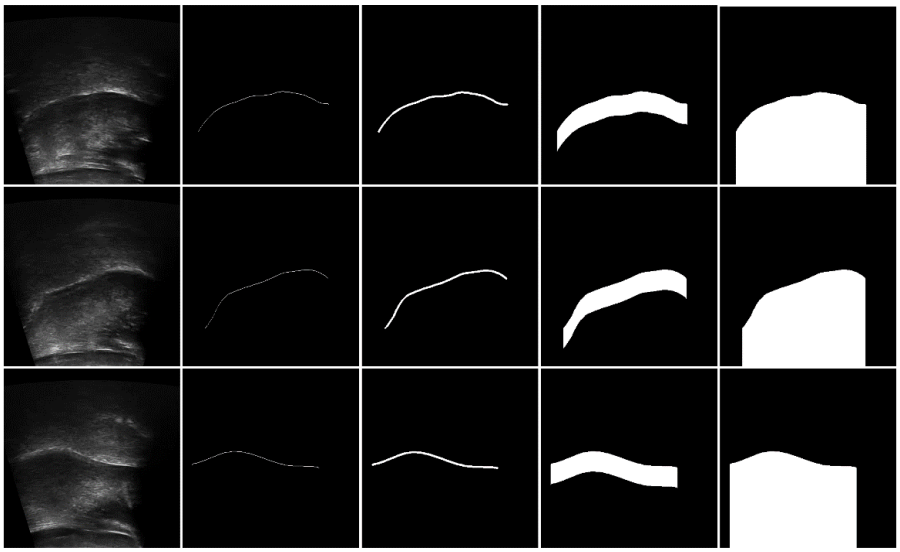


Figure 4-5. Samples of the OttawaSpeech dataset with different annotated tongue labels. From left to right, original ultrasound frame, mask with a diameter of one pixel, 5 pixels, 50 pixels, and the whole region under the tongue.

## 4.2 Face and Probe Tracking

To train the proposed deep learning model for the tacking of the ultrasound probe or 3D printable stabilizer device (details are explained in the following chapters), we made several annotated datasets. In our preliminary studies, we focused on using probe segmentation methods for localizing the ultrasound probe. Our results of these studies revealed that deep learning models such as sU-NET could track the probe when trained on a segmentation dataset with satisfying results. The occlusion issue was solved in many of these investigations. However, using ultrasound probe segmented regions, we could not find reliable calibration information as well as this technique was sensitive to the background color.

Therefore, although segmentation models are powerful for ultrasound probe and 3D stabilizer tracking in our preliminary studies, this method is not a robust alternative for an accurate tracking task. Two main issues of using image segmentation techniques for ultrasound probe tracking:

1. Non-accurate segmented regions impact the localization of the probe.

2. The segmentation task is more complicated than localization in terms of computational cost, results in a slower system.
3. Extracting location information needs a post-processing stage.

**Preliminary Investigations and Failure Cases:**

**Segmentation of the Whole Ultrasound Probe:** Creating a probe segmentation dataset is a time consuming and cumbersome progress because the probe region should be annotated pixel-by-pixel. We made a dataset of 500 images from our ultrasound probe. For the annotation process, we used the image processing toolbox of MATLAB, where the user can highlight regions of the probe easier using the mouse. The result of this method was significant in terms of accuracy, occlusion handling, and real-time performance. However, from ultrasound probe segmentation results, we could not grab robust information to calculate transformations required in the augmented reality module of our language pronunciation training system. Figure 4-6 illustrates an ultrasound probe segmented region in real-time. We trained our first deep learning model (sU-NET) for this experiment, while the occlusion handling of the model in real-time was significantly improved.

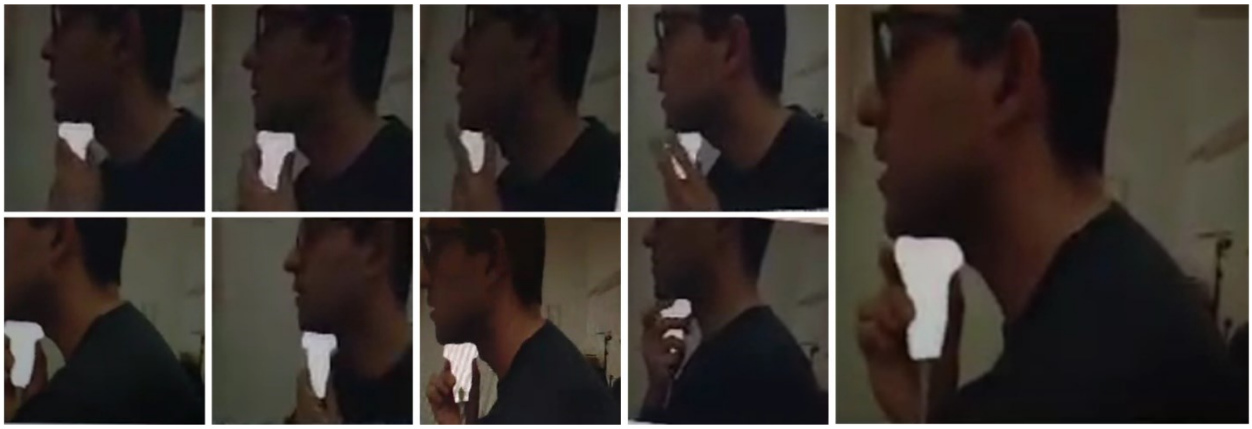


Figure 4-6. Randomly selected frames for the evaluation of real-time probe segmentation using the sU-NET deep learning model.

**Segmentation using a Bounding Box:** In order to find the location and orientation of the probe, we created another dataset comprises of two different ground truth labels, bounding boxes, and straight lines. The novelty of this work was that we kept one image of an ultrasound probe and a rectangle bounding box and a straight line (indicating the head of the probe) as the ground truth segmentation mask. Then we made data artificially by superimposing different artificial human hands and backgrounds. The

benefit of this method is that only one ground truth label is enough for all datasets. However, the accuracy of this method was low. Figure 4-7 shows some samples of this idea. The instances from different deep learning models, including original U-NET using this dataset was weak in terms of accuracy.

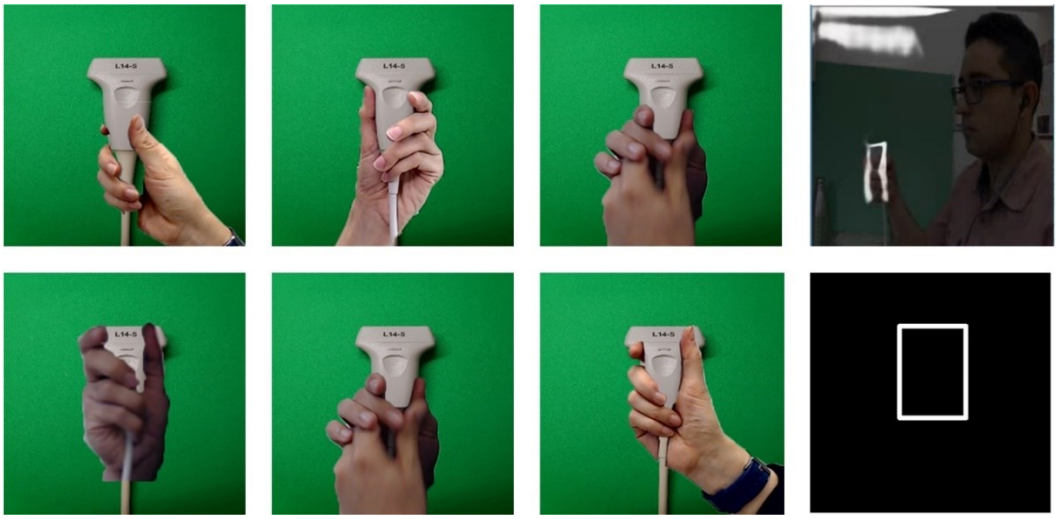


Figure 4-7. Using the bounding box as the ground truth label for real-time probe segmentation.

**Segmentation using Cross Lines:** We investigated the use of other segmentation masks, where the ultrasound probe location could be determined from segmentation masks. For instance, we created a dataset of images with annotated masks indicating the shortest and longest diameters (center of gravity for the probe). Figure 4-8 shows that the method is not reliable in terms of accuracy for probe detection and tracking in real-time.

Figure 4-8. Sample frames from real-time testing the sU-NET for segmentation of the probe for localization.

In general, our preliminary qualitative studies revealed that real-time tracking of the probe is not accurate using segmenting probe regions. We even investigated and created datasets for probe tracking using attaching a rectangular sticker on the probe head. In almost all methods, occlusion handling of deep learning models was significant, but the accuracy of the results is not applicable for tracking the probe location in real-time.

For this reason, we proposed a novel idea of tracking the ultrasound probe using a new deep learning model (named ProbeNet) designed for the calculation of location information of the probe. We used the ProbeNet for real-time tracking of the ultrasound probe (named freehand technique) *(Mohammad Hamed Mozaffari et al., 2019)* and a 3D printable device (called UltraChin) *(M Hamed Mozaffari & Lee, 2019)*, applicable for different ultrasound transducer types and shapes. Details of these methods are explained in the next chapters. Here, we focus on datasets that we used for the training of the ProbeNet.

The location and orientation of the ultrasound probe can be determined by having the position of two specified and fixed points on the probe. In two different approaches with and without using markers, we trained ProbeNet to track two defined points on the ultrasound probe and UltraChin in real-time. In the freehand approach, we trained ProbeNet by providing a dataset of images annotated by two endpoints of the probe head on a straight line. In this way, we could calculate location, orientation, and probe head size for each frame in real-time. The same approach was utilized for tracking the UltraChin device, while two pre-defined points are selected on two UltraChin markers.

**Freehand Ultrasound Probe Tracking Dataset:** Using our customized annotation software, a user places two points on the probe images. A straight line guides the annotator to select the correct positions on the two endpoints of the probe head easier.

In this dataset, we combined annotated images from two different kinds of data, images from real and artificial subjects. For the artificial dataset, we used a method of annotation, which will help researchers to annotate only one image (we named one-shot annotation). By keep the probe location in the rest of the images, while transforming other components of each image, we do not need to annotate the rest of the dataset image-by-image. In this way, after data augmentation, image and its corresponding ground-truth label are augmented automatically and provided new data. Figure 4-9 presents sample images from this real and artificial dataset.

As mentioned before, annotated data for each image are positions of two endpoints of the probe head. Therefore, for artificial images, we only need one annotated image, but for real images, we need to annotate each frame manually. We created 300 artificial images using Photoshop CS software (see Figure 4-9.d) as an artificial dataset. We superimposed four components, including head, hands, background, and ultrasound probe images, to create new images with different characteristics. For the real part of the dataset, we created 300 different frames from 5 different participants while they keep the ultrasound probe under their chin in mid-sagittal view (see Figure 4-9.c). Then, each frame was annotated by placing two pre-defined key points on each frame. In general, using this method, 600 randomly selected frames are annotated. Note that we made several annotated ultrasound tongue and ultrasound probe image datasets for each experiment in this thesis project, but we only report our optimum datasets for the sake of representation.

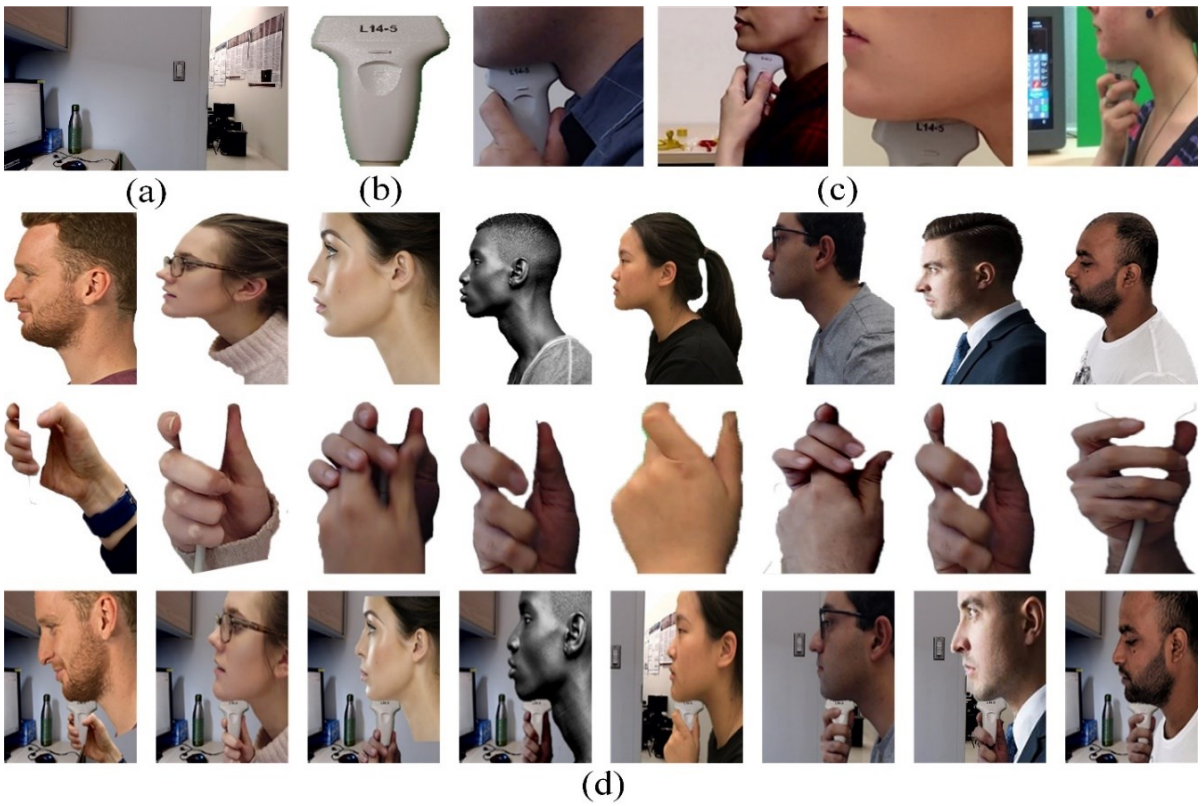



Figure 4-9. Samples of our training dataset for tracking of the ultrasound probe: (a) background image, (b) ultrasound probe image, (c) cropped images from real participants, (d) artificial images comprises of several different components (hand, head, background, ultrasound probe, and face).

# Chapter 5 Methods

In this chapter, we explain our proposed deep learning methods employed in two main modules of our second language pronunciation training system.

## 5.1 Automatic and Real-time Tongue Contour Tracking

### 5.1.1 sU-NET and sDeepLabV3

During the last decades, active contour models can be considered as the most common technique for tongue contour extraction and tracking. In this method, one curve should be drawn for initialization. Then, the initialized curve updates its location toward the tongue as the continuous brightest area in the image. Although this technique is effective, robust, and successful in many studies, due to its slow performance, dependency on initialization step, and many computational calculations such as gradient information, it cannot be used in fully automatic real-time applications. In recent years, deep learning techniques have been making significant advances in solving many problems that have not been solved for many years. Convolutional neural networks (CNNs), one of the most recent models in the deep learning field, present more robust and efficient results than their previous counterparts.

Tracking of the tongue contour in ultrasound video is a unique problem such that according to our experiments, the diversity of image distributions in different datasets is restricted to the flexibility and deformation states of the tongue muscle. It means that the tongue contour moves in ultrasound video data in specific regions with similar gestures and in the form of a bright gradient curve region. In other words, tongue ultrasound images that are captured from different ultrasound machines correlate in terms of image characteristics. Therefore, due to the similarity of the tongue contour in different datasets, it is feasible to have a general deep learning model applicable to the majority of tongue contour analysis applications.

Convolutional neural networks are usually used for the problem of image classification, where the output to an image is a single class label. Although image classification techniques provide valuable information for many medical image analysis studies *(Aminur et al., 2019; Sheikh Hassani & Green, 2019)*, the desired output is usually an image of the location and size of a lesion. Considering this reason, we proposed a simple deep learning method, employing CNN layers, inspiring from the architecture of the FCN *(Long et al., 2015)*, the SegNet *(Badrinarayanan et al., 2015)*, and the U-NET

*(Ronneberger et al., 2015)* models. Our proposed model is utilized for automatic real-time tongue contour tracking in our designed preliminary language training system.

In models like U-NET *(Ronneberger et al., 2015)* or SegNet *(Badrinarayanan et al., 2015)* architecture, there are several layers in each stage of encoding and decoding with a huge number of parameters. Although many trainable parameters in a network might result in a better regularization, it needs more data for training with the expense of bigger memory and computational cost. We found that omitting several convolutional layers in each stage of encoder and decoder blocks improves architecture performance in terms of speed. The network accuracy still is comparable with the original model for the small-sized images, while the receptive field of the network decrease by omitting each layer of the network. In this study, we applied a modified version of popular CNN model for biomedical applications *(Falk et al., 2019)*, U-NET *(Ronneberger et al., 2015)*, for the problem of tongue segmentation.

We proposed a simplified version of U-NET (called sU-NET) through decreasing network size and the number of filters. For instance, sU-NET has 14 layers in total instead of 23 layers of the original U-NET model by omitting one layer in each step of the encoding-decoding process. As a consequence, sU-NET *(Hamed Mozaffari et al., 2019; M. Hamed Mozaffari et al., 2018)* can automatically delineate the tongue contour from ultrasound long video data in real-time applications. According to the recent study in the field of ultrasound tongue analysis and tracking *(Laporte & Ménard, 2018)*, there have been few attempts to apply CNN architectures for ultrasound tongue datasets. This thesis study is the first example of employing CNN for automatic tongue tracking as well as using CNN in ultrasound-guided second language training applications.

Figure 5-1 illustrates our proposed architecture. sU-NET consists of repeated 3×3 convolutions with no zero paddings, each followed by a rectified linear unit (ReLU). A 2×2 max pooling operation with a stride of 2 is applied in each down-sampling step. In the up-sampling path, each layer consists of a deconvolutional layer followed by a convolutional layer. For each level of the up-sampling path, the corresponding feature map from the encoder block is cropped and concatenated with the results of the up-sampling layers. At the final layer, a 1×1 convolution is used to find the desired segmentation image. For a better understanding of sU-NET layers and their connections, see Figure 5-1. As can be seen from the figure, the output feature map is smaller than the input image. Using zero paddings, sU-NET can be trained on datasets to provide the same resolution images in the output and input layers.

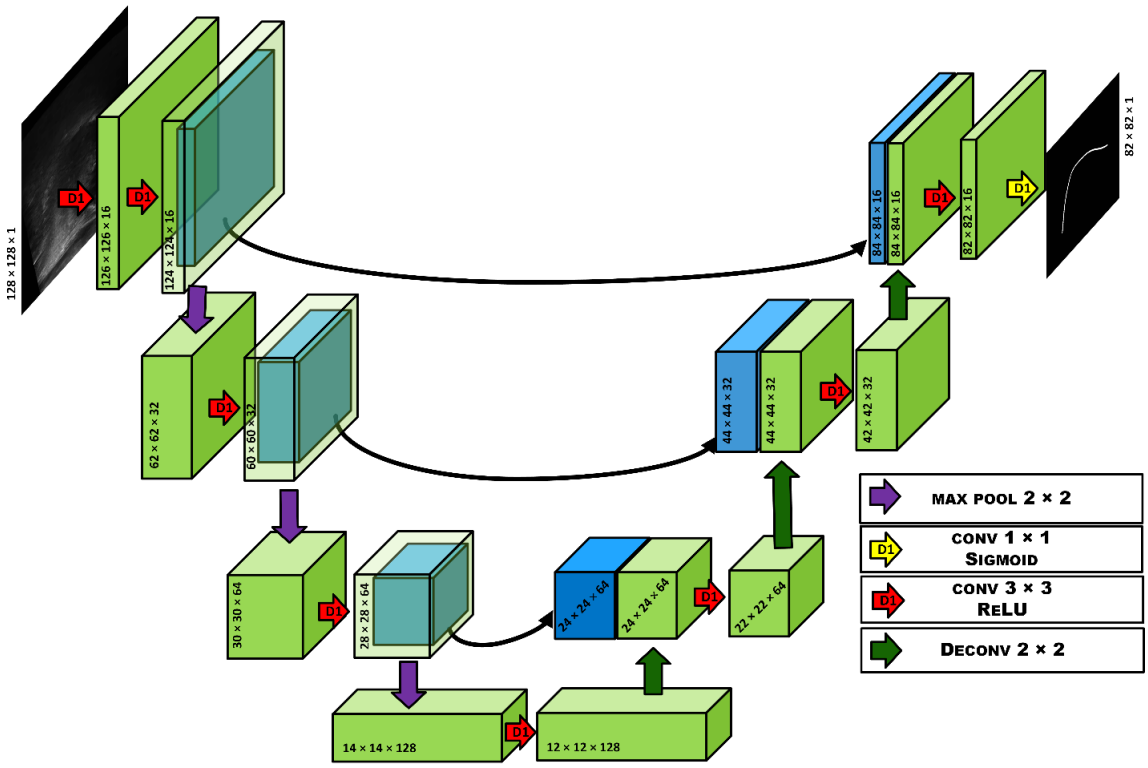



Figure 5-1. sU-NET architecture. Each cube corresponds to a multi-channel feature map and different arrows color illustrate the different operations. The numbers near or inside each box are feature map size and depth.

### 5.1.2 BowNet and wBowNet

Our preliminary deep learning model (sU-NET) works fast for tongue contour tracking, but its ability in over-fitting handling is not significant. Furthermore, sU-NET performs well on small-sized images of the tongue, similar to the previous deep learning approaches *(Fasel & Berry, 2010)*. For this reason, we attempt to design specific models without compromising the real-time performance of sU-NET. In general, we attempt to design an end-to-end deep learning model using CNNs, applicable for every arbitrarily sized ultrasound tongue dataset. At the same time, it must work automatically, fast, with higher accuracy.

For this reason, we embedded sU-NET in two parallel and interconnected architectures that benefit from dilated convolutions. We first modified one of the advanced models in semantic segmentation, DeepLabV3 *(L.-C. Chen et al., 2017; Hamaguchi et al., 2018)*. DeepLabV3 uses dilated convolutions to keep the receptive field of the network larger, while fewer trainable parameters are used in the network. We designed sDeepLabV3, which is a simplified version of the DeepLabV3 model. From combining our proposed encoder-decoder model (sU-NET) and dilated CNN model (sDeepLabV3), we

proposed two new deep neural network models named BowNet (see Figure 5-2). It is noteworthy that this idea of using two networks in parallel was published recently by google company for semantic segmentation tasks *(L.-C. Chen et al., 2018)*.

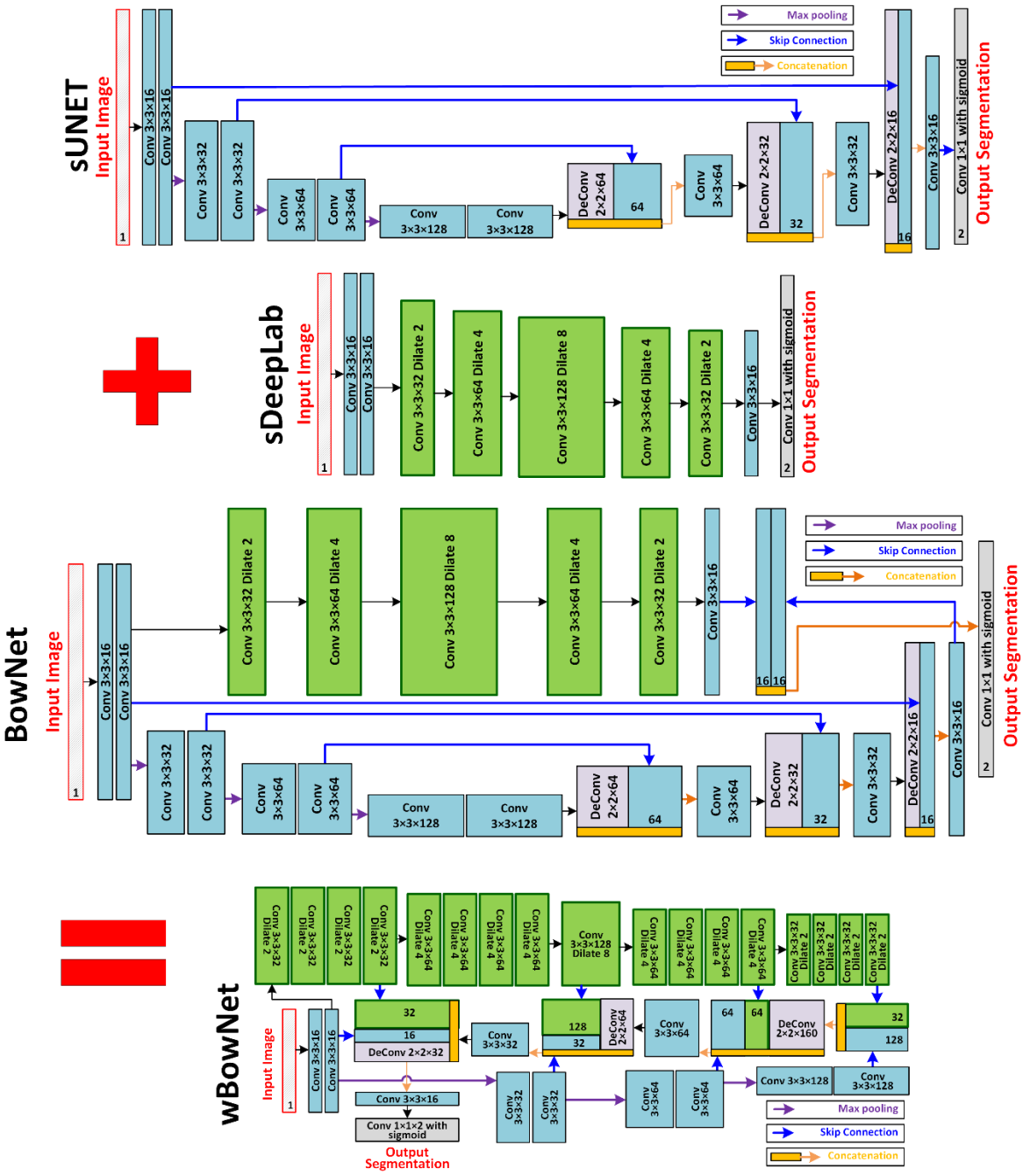



Figure 5-2. BowNet architectures from a combination of sDeepLabV3 and sU-NET models.

The BowNet architecture has two separate sub-network forward paths, whereas the results of the two paths are concatenated at the last layer, followed by a fully convolutional layer. On the contrary, the wBowNet has two interconnected (weaved) sub-networks. Figure 5-3 and Figure 5-4 illustrate the BowNet and wBowNet network architecture, respectively. Details of each neural network model used in this study are listed in Table 5-1. It is vital to pay attention to both segmentation context and resolution for detecting tongue contours in relatively noisy ultrasound images. In this application, context can be extracted using dense classification methods *(M. Hamed Mozaffari et al., 2018)*. Nevertheless, up-sampling methods such as deconvolution are not able to recover the low-level visual features which are lost in the down-sampling stage *(G. Lin et al., 2017)*, resulting in low-resolution segmented instances.

BowNet models consider both localization and globalization detection in their architecture using two different receptive fields of standard and dilated convolutions. It can be seen from Table 5-1 that all the deep convolutional networks in this study can be trained and tested in an end-to-end fashion. It means that networks accept an image as input and provides the output instance as a probability map, directly. In each network architecture, convolutional layers use ReLU activation as a non-linearity function. Dropout layers and batch normalization layers are followed by convolutional layers to improve the regularization, convergence speed, and accuracy of network models. From Table 5-1, dilation factor is first increased from one to eight and then decreased again to one in a forward path of successive dilated convolutions to solve the problem of spatial inconsistency *(Hamaguchi et al., 2018)*.

It is common in deep learning semantic segmentation to use SoftMax for the last layer of the network, and at least part of the system is a modified version of the VGG16 network *(Badrinarayanan et al., 2015; Long et al., 2015; Simonyan & Zisserman, 2015)*. Moreover, multiclass cross-entropy is employed as a loss function for the training of networks. Ultrasound datasets usually contain gray-scale images, and the desired target for tongue contour extraction has only two class labels of tongue contour and background. For this reason, we optimized binary cross-entropy as the loss function, and a fully convolutional layer, along with a sigmoid activation function is utilized in the last segment of all the proposed networks.

To fully leverage the power of the trained network for ultrasound tongue contour segmentation, unlike many similar studies with small images as the output *(Fasel & Berry, 2010; Jaumard-Hakoun, Xu, Roussel-ragot, et al., 2015)*, we keep the image sizes in both input and output layers as $128 \times 128$ pixels. Input images are cropped and scaled to make

them square for the sake of applying convolutional layers, and zero-padding is used to keep the image size throughout the network.

Table 5-1. The network architecture of BowNet, wBowNet, sU-NET, and sDeepLabV3. In convolution layers, ConvYDX, Y is the number of kernels, and X is the value of the dilation factor. FCL=Fully Convolutional Layer.

| | NL | wBowNet | | NL | BowNet | | NL | sU-NET | NL | sDeepLabV3 |
|---|---|---|---|---|---|---|---|---|---|---|
| INPUT 128 × 128 | 1 | GRAYSCALE | | 1 | GRAYSCALE | | 1 | GRAYSCALE | 1 | GRAYSCALE |
| | | Path-1 | Path-2 | | Path-1 | Path-2 | - | - | - | - |
| CONV-1 | 2 | Conv16 | - | 2 | Conv16 | - | 2 | Conv16 | 2 | Conv16 |
| POOL-1 | 1 | MaxPool | - | 1 | MaxPool | - | 1 | MaxPool | - | - |
| CONV-D2 | 4 | - | Conv32D2 | 1 | - | Conv32D2 | - | - | 1 | Conv32D2 |
| CONV-2 | 1 | Conv32 | - | 1 | Conv32 | - | 1 | Conv32 | - | - |
| POOL-2 | 1 | MaxPool | - | 1 | MaxPool | - | 1 | MaxPool | - | - |
| CONV-D4 | 4 | - | Conv64D4 | 1 | - | Conv64D4 | - | - | 1 | Conv64D4 |
| CONV-3 | 1 | Conv64 | - | 1 | Conv64 | - | 1 | Conv64 | - | - |
| POOL-3 | 1 | MaxPool | - | 1 | MaxPool | - | 1 | MaxPool | - | - |
| CONV-D8 | 1 | - | Conv128D8 | 1 | - | Conv128D8 | - | - | 1 | Conv128D8 |
| CONV-D4-2 | 4 | - | Conv64D4 | 1 | - | Conv64D4 | - | - | 1 | Conv64D4 |
| CONV-D2-2 | 4 | - | Conv32D2 | 1 | - | Conv32D2 | - | - | 1 | Conv32D2 |
| CONV-4 | 1 | Conv128 | - | 1 | - | Conv16 | - | - | 1 | Conv16 |
| CONV-5 | - | - | - | 1 | Conv128 | - | 1 | Conv128 | - | - |
| CONCATE & CROP | 1 | CONV-4, CONV-D2-2 | | - | - | - | - | - | - | - |
| UP-CONV-1 | 1 | Transpose-Conv | | 1 | Transpose-Conv | - | 1 | Transpose-Conv | - | - |
| CONCATE & CROP | 1 | UP-CONV-1, CONV-3, CONV-D4-2 | | 1 | UP-CONV-1, CONV-3 | - | 1 | UP-CONV-1, CONV-3 | - | - |
| CONV-6 | 1 | Conv64 | | 1 | Conv64 | - | 1 | Conv64 | - | - |
| UP-CONV-2 | 1 | Transpose-Conv | | 1 | Transpose-Conv | - | 1 | Transpose-Conv | - | - |
| CONCATE & CROP | 1 | UP-CONV-2, CONV-2, CONV-D8 | | 1 | UP-CONV-2, CONV-2 | - | 1 | UP-CONV-2, CONV-2 | - | - |
| CONV-7 | 1 | Conv32 | | 1 | Conv32 | - | 1 | Conv32 | - | - |
| UP-CONV-3 | 1 | Transpose-Conv | | 1 | Transpose-Conv | - | 1 | Transpose-Conv | - | - |
| CONCATE & CROP | 1 | UP-CONV-3, CONV-1, CONV-D2 | | 1 | UP-CONV-3, CONV-1 | - | 1 | UP-CONV-3, CONV-1 | - | - |
| CONV-8 | 1 | Conv16 | | 1 | Conv16 | - | 1 | Conv16 | - | - |
| CONCATE & CROP | - | - | | 1 | CONV-8, CONV-4 | | - | - | - | - |
| OUTPUT 82 × 82 | 1 | FCL | | 1 | FCL | | 1 | FCL | 1 | FCL |

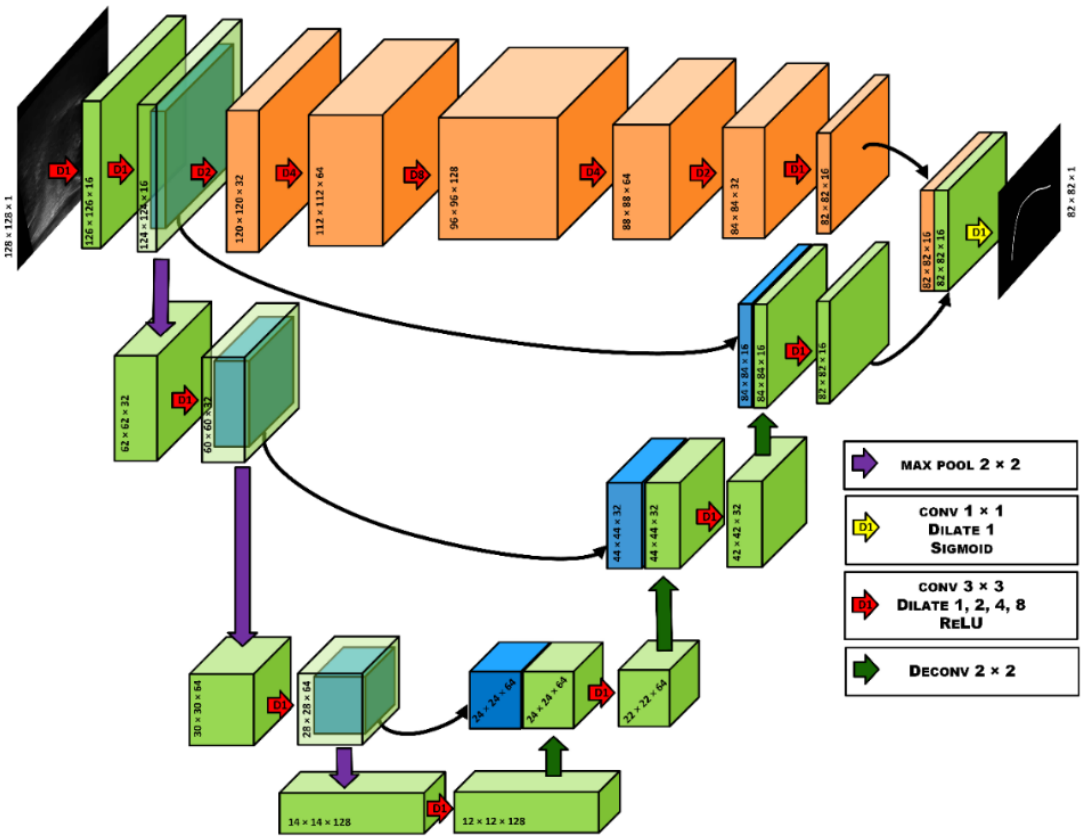



Figure 5-3. Overview of the proposed BowNet architecture. In each layer, filter kernels are depicted using boxes. The green and orange boxes are results of standard and dilated convolution layers, respectively.

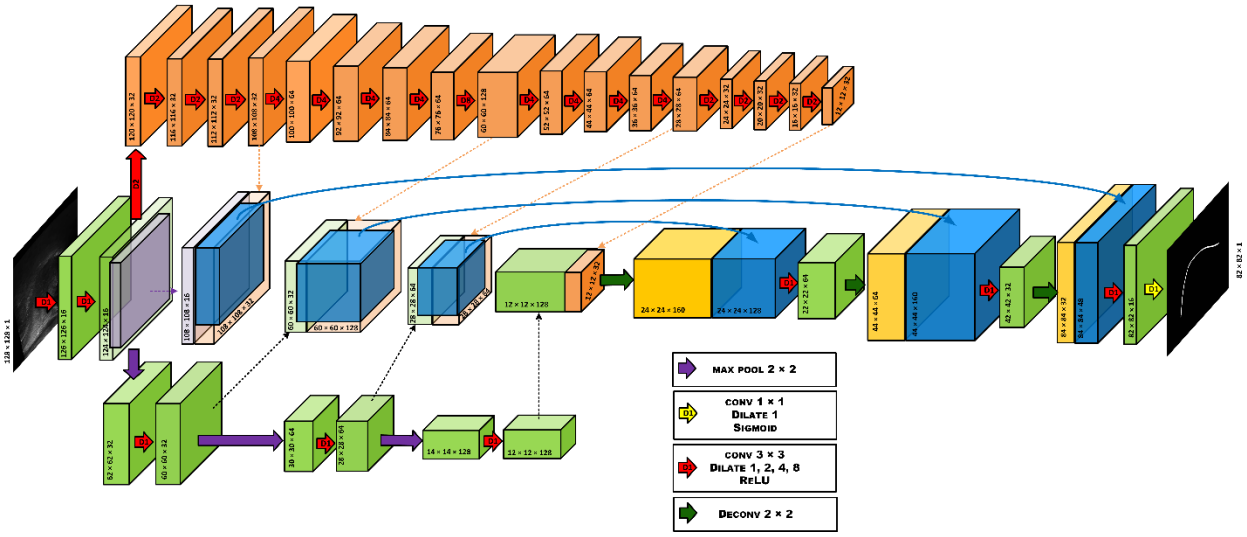



Figure 5-4. Overview of the proposed wBowNet architecture. In each layer, filter kernels are depicted using boxes. The green and orange boxes are results of regular and dilated convolution layers, respectively.

Some random feature maps of each layer in the BowNet network during the training process are shown in Figure 5-5. The sparsity of the dilated convolutional layer causes a bigger receptive field, whereas the result might be the false detected area as the tongue contour, but with more accurate shapes. The checkerboard artifact can be seen clearly in up-sampling layers where deconvolutional layers are applied to the previous feature maps. Summation of both dilated and regular convolution layers results in a uniform contour region in the output results.

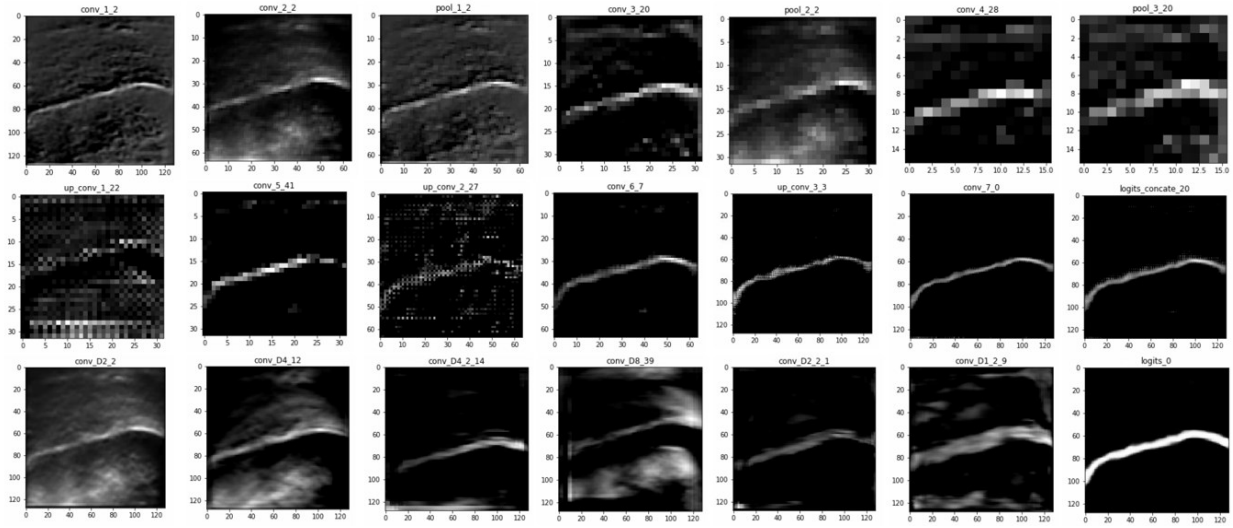



Figure 5-5. Random feature maps from different layers of the BowNet. Two rows are related to the encoder-decoder forward path, and the last row is related to the dilated consecutive dilated convolutions. The checkerboard effect in the up-sampling section can be seen clearly.

### 5.1.3 IrisNet and Peripheral Vision

By checking almost any deep learning competition leaderboards such as *(Web, 2019)*, state-of-the-art models for semantic segmentation including PSPNet *(H. Zhao et al., 2017)*, DeepLab V3+ *(L.-C. Chen et al., 2018)*, to name few have not been used for other tasks, including lesion segmentation or medical image segmentation. From *(S. Liu et al., 2019)*, the main reason is that the semantic segmentation models are designed with the assumption of using optimized encoders pre-trained on a large dataset with a similar domain. Table 5-2 shows that almost all advanced deep learning semantic segmentation models are based on a backbone model pre-trained on a large dataset.

Table 5-2. Several examples of advanced semantic segmentation methods. Each model uses a backbone model pre-trained on a dataset. Some models have several backbones (**) and pre-trained on several datasets (*).

| **Model** | **Backbone model** | **Pre-trained Dataset** |
|---|---|---|
| ACE *(Z. Wu et al., 2019)* | VGG16 | ImageNet |
| CCNet *(Huang et al., 2019)* | ResNet | ImageNet |
| BowNet *(M. Hamed Mozaffari et al., 2019)* | N/A | N/A |
| Gated-SCNN *(Takikawa et al., 2019)* | ResNet | Ms-COCO |
| RTSEG *(Siam et al., 2018)* | ResNet** | ImageNet |
| DenseASPP *(Yang et al., 2018)* | ResNet** | ImageNet |
| LRefineNet *(Nekrasov et al., 2018)* | ResNet** | ImageNet* |
| DeepLabV3+ *(L.-C. Chen et al., 2018)* | ResNet** | MS-COCO |
| DeepLabV3 *(L.-C. Chen et al., 2017)* | VGG16 | MS-COCO |
| DeepLabV2 *(L.-C. C. Chen et al., 2017)* | ResNet** | ImageNet |
| RefineNet *(G. Lin et al., 2017)* | ResNet | MS-COCO* |
| SegNet *(Badrinarayanan et al., 2015)* | VGG16 | ImageNet |
| PSPNet *(H. Zhao et al., 2017)* | ResNet | MS-COCO |
| LinkNet *(Chaurasia & Culurciello, 2017)* | ResNet** | ImageNet |
| FPN *(T.-Y. Y. Lin et al., 2017)* | ResNet** | MS-COCO |
| FCRNs *(Laina et al., 2016)* | ResNet | MS-COCO* |
| Attention *(L.-C. Chen et al., 2016)* | DeepLab | ImageNet** |
| ENet *(Paszke et al., 2016)* | ResNet | MS-COCO* |
| U-NET *(Ronneberger et al., 2015)* | N/A | N/A |
| DeconvNet *(Noh et al., 2015)* | VGG16 | ImageNet |
| FCN8 *(Long et al., 2015)* | VGG16 | ImageNet |
| DeepLabV1 *(L.-C. Chen et al., 2014)* | VGG16 | ImageNet |

For this reason, variants of the U-NET model are still the best models in many medical image analysis tasks (see *("medicaldecathlon," 2019; "paperswithcode," 2019)*. Our investigation on image segmentation in many fields of science indicates that a general optimized pre-trained model is required for high accuracy instances. For this reason, models such as the original U-NET are prevailing in different fields of science than advanced segmentation models, because no pre-trained model is necessary as an encoder block. For medical image segmentation, there is still room for improvement, and we believe IrisNet could be a promising alternative method.

From our experimental results, BowNet architectures revealed significant improvement in performance for the problem of tongue contour tracking in ultrasound data. However, the generalization ability of these techniques for other ultrasound tongue datasets is weak. To address this issue, we studied the effect of domain adaptation in tongue contour tracking in ultrasound video for the first time. Our assessments of U-NET

and DeconvNet models indicate that adapting domain adaptation techniques improves the results of deep learning models *(Hamed Mozaffari & Lee, 2019)*. However, lack of models, pre-trained on a large ultrasound tongue dataset, as well as non-compatibility of the current well-known pre-trained models such as VGG *(Simonyan & Zisserman, 2015)* and ResNet *(He et al., 2016)* for BowNet structures, motivates us to focus on optimizing a deep model instead of using transfer learning approach *(S. Liu et al., 2019)*. It is noteworthy to repeat that the main reason for the significant achievements of advanced deep learning models in the semantic segmentation field is because of using encoder models, pre-trained on a huge dataset. For the medical ultrasound image analysis field, there is still no such a massive and general dataset to use in the pre-training stage *(S. Liu et al., 2019)*. For this reason, the best alternative for current research progress is to optimize networks or using specific smaller models for each dataset *(S. Liu et al., 2019)*.

Employing down-sampling techniques in deep learning models such as max-pooling layers has resulted in better contextual predictions in many major computer vision tasks such as image classification and detection *(Ronneberger et al., 2015)*. In the field of image segmentation, a desirable result should contain accurately delineated regions with a contour around the target object. Due to the loss of information, down-sampling provides lower resolution prediction maps, which is not desirable for image segmentation tasks. Omitting pooling layers and replacing them with new operations such as dilated convolutions in the semantic segmentation field is a new idea. Unlike down-sampling, dilated convolution can keep the receptive field of a deep learning model with the same resolution of the input image *(L.-C. Chen et al., 2017)*.

In many recent publications, dilated convolution outperformed encoder-decoder techniques but with introducing grinding artifact to the results *(Yu et al., 2017)*. Recent studies showed that using feature maps from different layers of a network in the shape of encoder-decoder increases the performance of a model *(L.-C. Chen et al., 2018)*. Therefore, benefiting from both encoder-decoder style and dilated architecture, novel models could find better image segmentation results *(L.-C. Chen et al., 2018)*. Nevertheless, in the field of medical image analysis, U-NET style architectures have still been popular and outperformed other techniques *(Altaf et al., 2019; Falk et al., 2019; Litjens et al., 2017; Sudheer Kumar & Shoba Bindu, 2019)*.

Recently, specific deep learning networks have been utilized for the problem of ultrasound tongue contour tracking *(M. Hamed Mozaffari et al., 2018, 2019)*. The generalization ability of those methods is also investigated for other datasets *(M. Hamed Mozaffari & Lee, 2019a)*. However, fine-tuning is a vital step for using a model to work

on a novel dataset. Note that negative transfer is another issue for using a pre-trained model for different datasets *(S. Liu et al., 2019; M. Hamed Mozaffari & Lee, 2019a)*. The generalization of a deep learning model for image segmentation is not clear, and the best method for a deep learning model is fine-tuning for only the imagery of a specific image context *(Guo et al., 2018)*.

Fortunately, different ultrasound tongue datasets have similar characteristics such as point of view, which is often mid-sagittal cross-section, bright thick line in specific regions of the image (around 8 cm up from the surface of the chin *(M. Li et al., 2005)*), and almost similar image resolutions. Furthermore, possible gestures of the tongue are limited to the alphabet and vocabulary range of the speaker's language *(Bliss, Bird, et al., 2018)*. Moreover, speckle-noise patterns and artifacts are almost analogous for different subjects due to the limited movements of ultrasound transducer during data acquisition, relatively similar human oral region, and stable reference points such as palate or jaw hinge in datasets. Therefore, we attempt to design a general and accurate deep learning model applicable for almost all standard ultrasound datasets, with the capability of real-time performance without any fine-tuning or image pre- or/and post- enhancements.

**RetinaConv:** The procedure of the correct segmentation task by the human brain has been unclear for researchers *(Guo et al., 2018)*. However, we know that the human eye has the ability of peripheral vision. One crucial strength of the human eye is to detect objects and movements outside of the direct line of sight, away from the center of gaze. This ability, called peripheral (side or indirect) vision, helps us to detect and sense objects without turning our head or eyes, resulting in less computation for our brain.

Our vision around the central part of our eye's field of view is sharper than far from the center. Following the human eye's peripheral vision ability, we designed a new convolutional module named RetinaConv. We simulated the idea of peripheral vision in the human eye by a convolutional filter module that is illustrated in Figure 5-6, where the center of the filter is stronger than around. One might consider this filter as a Gaussian filter as a combination of two kernels.

To make the RetinaConv filter module, we utilized the distributivity property of the convolution operator: $f * (g + h) = f * g + f * h$ where $f$ is the input image, $g$ and $h$ are standard and dilated convolutional filters, respectively. Therefore, applying two filters is equivalent to using the summation of them. By changing filter size and dilation factor, different peripheral vision strengths can be achieved for different sized images during the hyperparameter tuning stage. The benefit of RetinaConv is not limited to merely

computational speed, but also to accuracy enhancement due to the use of two receptive fields simultaneously.

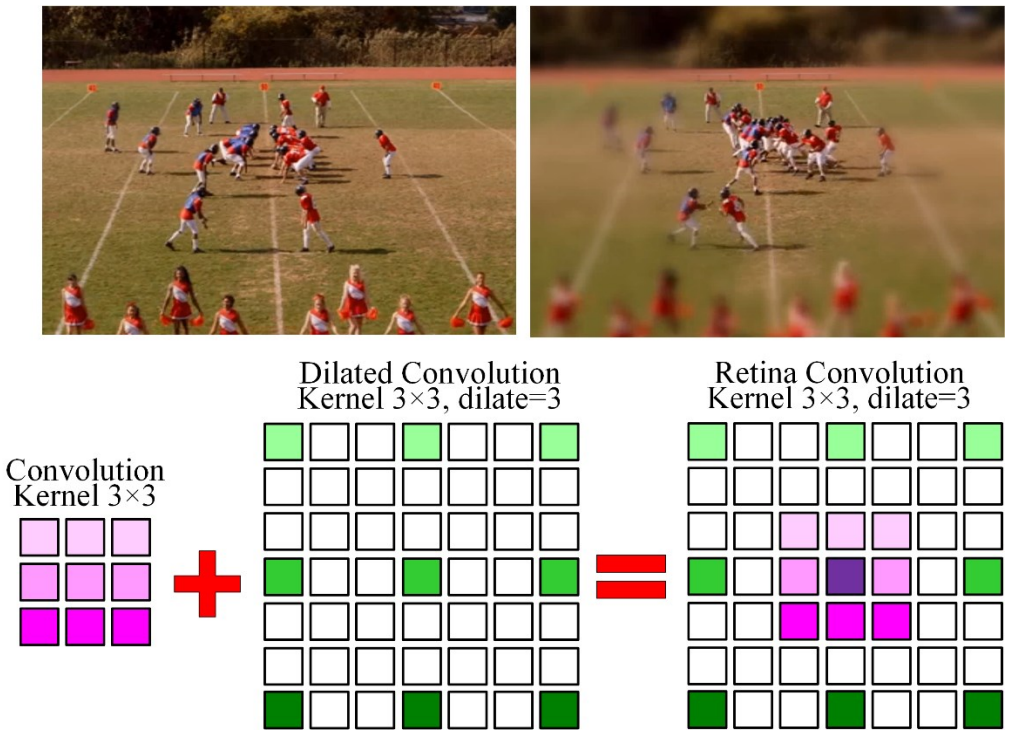



Figure 5-6. Example of peripheral vision in the human eye. Centre of gaze is sharper due to more light detectors on Retina (dense kernel or standard convolution) and around is blurry because of fewer detectors on Retina (sparse kernel or dilated convolution).

Deep learning models such as DeepLab, with different versions (v1 *(L.-C. C. Chen et al., 2017)*, v2 *(L.-C. C. Chen et al., 2017)*, v3 *(Szegedy et al., 2016)*, v3+ *(L.-C. Chen et al., 2018)*) proposed by Google company, are the current robust networks which have been shown to be effective in semantic segmentation tasks for the context of natural images. Using DeepLab for ultrasound tongue imaging requires a pre-trained DenseNet model trained in advance on a huge source grey-scale ultrasound dataset (not in RGB format). A modified version of DeepLab v3 *(Szegedy et al., 2016)* without pre-trained weights was tested for tongue contour segmentation *(M. Hamed Mozaffari et al., 2019)*, the result was not significant. DeepLab models are also huge networks in terms of parameters that require a robust training and testing system. On the other hand, U-NET *(Ronneberger et al., 2015)* is a ubiquitous model for medical image segmentation outperforming many other models *(Falk et al., 2019)* without the usage of pre-trained weights.

To maintain the powerful performance of deep learning models simultaneously, such as DeepLab and U-NET, we designed the IrisNet network. As an encoder-decoder structure like U-NET, IrisNet use extracted features from encoder layers passed to the decoder layers as well as max-pooling operator. However, the RetinaConv module in each layer keeps the receptive filed wider by utilizing dilated convolution. Individual kernels of RetinaConv highlights mid-point regions more by emphasizing the surrounding area. As a combination of kernels, grinding *(Yu et al., 2017)* and checkerboard artifacts *(Odena et al., 2016)* due to the inappropriate filter sampling rate are alleviated significantly.

However, analogous to a Gaussian filter, RetinaConv provides more blurry features. To address this issue, we used a dilation factor of 1 on both end sides of the network. The network architecture of IrisNet is presented in Figure 5-7. The figure illustrates the RetinaConv module, encoder, and decoder blocks, as well. Using different operators such as max pooling, transpose convolution, skip connections, and RetinaConv, IrisNet able to predict delineated regions by using both low- and high-level features at the same time results in better segmentation output.

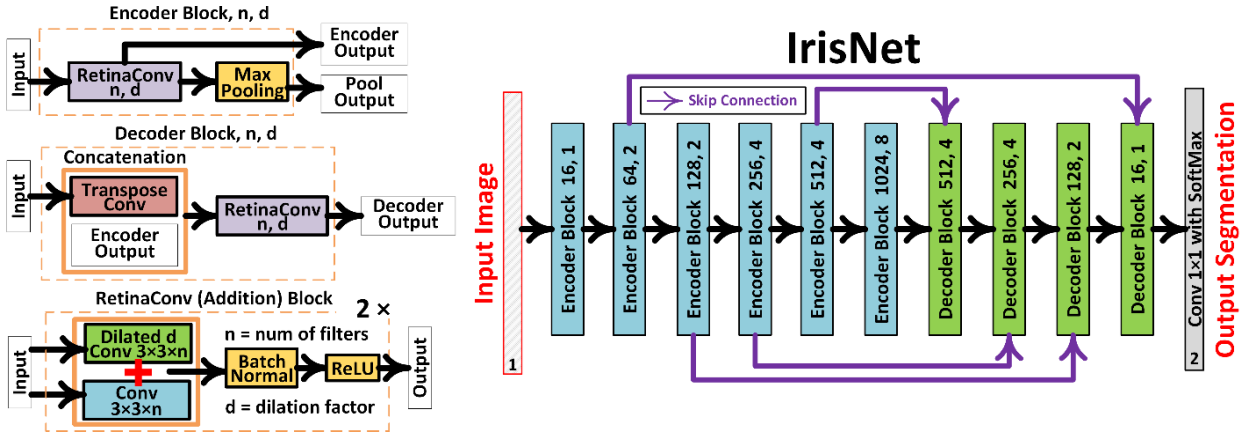



Figure 5-7. Network Architecture of IrisNet. RetinaConv module is used in both encoder and decoder layers each for two times. The plus symbol is the sum operator, while the squared box around two kernels indicates a concatenation operator.

Each encoder and decoder block in IrisNet have two times repeated RetinaConv blocks following by batch normalization (BN), ReLU activation function, and max-pooling layers. The number of filters and dilated factors are indicated in Figure 5-7. Within each decoder block, feature maps from the previous layer first are upsampled using 2×2 transposed convolution. Then the output is concatenated by the result of the corresponding encoder block skipped from the later encoder section. In the last layer, we used a 1×1, fully convolutional layer.

In former deep learning models, which have been proposed for automatic tongue contour tracking *(Fasel & Berry, 2010; M. Hamed Mozaffari & Lee, 2019a; J. Zhu et al., 2019)*, the sigmoid activation function in the output layer provides on channel grey-scale instances. Therefore, ground truth labels should be grey-scale with 255 classes after data augmentation. Following semantic segmentation literature, we developed the IrisNet model to predict instances comprises of two binary channels, one for foreground and another for background labels. Therefore, the last layer activation function in IrisNet is SoftMax, and ground truth labels are in a binary format even after data augmentation.

Although this method of using two classes is not a new idea for the medical image segmentation field *(Ronneberger et al., 2015)*, it is a novel idea for the ultrasound tongue contour tracking. The importance and impact of using background information in the network training stage are that the model can also learn which area of an image belongs to the background. See Figure 3-1 for an example of artifacts that can be detected as a tongue surface but its artifact shadow from the jaw. Therefore, IrisNet can easier discriminate background artifacts and noise from the region of interest. Furthermore, this technique releases researchers from cropping extra information such as ultrasound settings, dataset brand, or palate/jaw regions. Our experimental results also revealed the importance of using two classes instead of one in this literature.

## 5.2 Ultrasound Probe Tracking

The purpose of the ultrasound probe tracking module is to find real-time information about location, orientation, and scaling of the probe to calculate transformation data (calibration) to map the corresponding ultrasound frame and segmented tongue contour on the face of the user. Consequently, as a simple augmented reality application, language learners can observe their tongue movements, with highlighted contour region, on their face side view.

**Preliminary study:** In the first attempt, we investigated the use of the face profile tracking of a language learner as the clue for transforming ultrasound video on the user's face view. Primary modules of the pilot language pronunciation training system are illustrated in Figure 6-25. As can be seen in the figure, there are two modules, off-line for recording, and online for real-time language training. In both modules, the ultrasound data, contain tongue contour information extracted by our proposed tracking method, are overlaid automatically on a video recorded from face profile.

In general, two superimposed videos are played on a computer's screen for a language learner, one from a real-time recording of him/his tongue and the other from an instructor's tongue during speech. The first system of pronunciation training using ultrasound was entirely off-line *(Abel et al., 2015; Yamane et al., 2015)*, where video data should be manipulated using editing software. Furthermore, language learners could not see their tongue movements in real-time, and overlaid videos are a presentation of two videos superimposed on each other. Therefore, two videos should be recorded consequently from one user while his or her head and ultrasound probe is fixed in both videos *(Abel et al., 2015; Bird et al., 2018)*. Our preliminary investigation might be considered as an updated version of this system *(M. Hamed Mozaffari et al., 2018)*, where the user can move her or his head and ultrasound probe.

Moreover, ultrasound tongue contours are illustrated to the user as guidance. In this system, to find the best position for overlaid video, we used Haar cascade method in the OpenCV library *(Viola & Jones, 2004)*, where face profile of the user is tracked, and ultrasound video moves with the calibration information from the tracking method. Results of our first system indicate that tracking the face profile is not a reliable method because of many issues such as occlusion, slow tracking, non-accurate superimposing, and manual works as well as non-real-time system. For this reason, we employ a deep learning technique to track a specific object instead of a user's face. The following are our two approaches to solving these issues.

### 5.2.1 Freehand Tracking of the Ultrasound Probe

In previous ultrasound-enhanced multimodal studies, the head of a language learner should be stabilized during video and ultrasound data collection *(Abel et al., 2015; Bird et al., 2018)*. This obligation is for ensuring that ultrasound videos are controlled with respect to the orientation of the probe as well as providing a positional reference in quantitative assessments *(Bird et al., 2018)*. A consequence of this restriction is the reduction of the flexibility of the head's movement. Moreover, the learner should concentrate on the user interface for a long time with limited movements, which might result in body fatigue and eye strain, ultimately reducing the effectiveness of the L2 training session.

In a few recent studies *(M. Hamed Mozaffari et al., 2018; Mohammad Hamed Mozaffari et al., 2019)*, there is no obligatory requirement for stabilization of a language learner's head due to the use of automatic face tracking, results in flexibility for the head location as well as a freehand placement of the ultrasound probe (see Figure 5-8 for some

samples of stabilization methods). However, in this method, different face profiles (different genders, skin characteristics, and ages) might provide different results, and ultrasound video frames are overlapped on the face with some non-accuracy. In order to make our system independent from the user's face profile and make it a universal toolkit for any users, we track the ultrasound transducer instead of the face. Besides the face profile independent characteristic of this approach, tracking of the probe also provides reference points, which are required for transforming ultrasound video frames to overlay on the user's face. Our experimental assessments revealed that our deep learning tracking method could be applied to several types of ultrasound transducers as a universal technique.

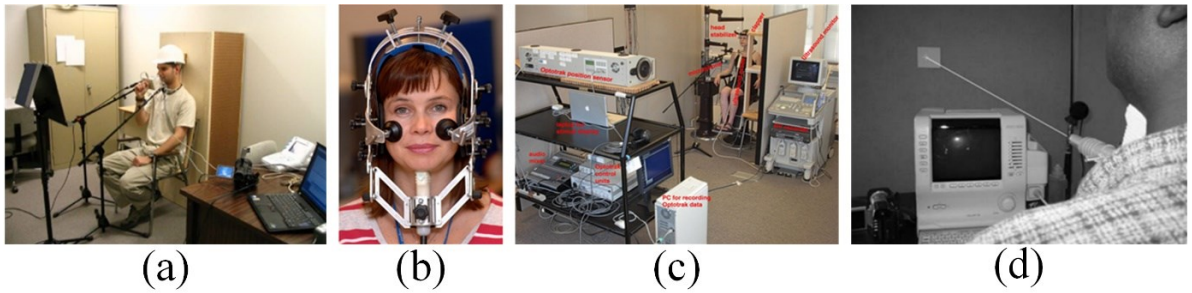



Figure 5-8. An illustration of some stabilization methods for head and probe. a) helmet fixed to the wall *(Ménard & Noiray, 2011)*, b) designed helmet keep the probe under the chin *(Scobbie et al., 2011)*, c) optical tracking system for the head alignment *(F. Campbell et al., 2010)* and d) a laser connected *(Gick, 2002)* to the probe detects movements of the neck.

Object localization (and/or detection) is determining where objects are located in a given image using bounding boxes encompass the target object. Various deep learning methods have been proposed for object localization in recent years *(Yilmaz et al., 2006b; Z.-Q. Zhao et al., 2018)*. Similar to facial landmark detection *(Y. Wu & Ji, 2019)*, when several key points of the human face are detected as landmarks, we defined two key points on ultrasound transducer for the sake of probe tracking. The ultrasound probe is tracked automatically in real-time using our new deep convolutional neural network (we named that ProbeNet). In this method, positions of the two key points provide us valuable information in each frame, comprises of probe orientation, location, and a reference for the ultrasound image scale. We designed ProbeNet for the probe tracking problem by inspiring from VGG16 architecture *(Simonyan & Zisserman, 2015)*.

Tracking of an ultrasound probe has already been accomplished using different kinds of devices such as electromagnetic, optical, GPS, and mechanical sensors *(Mohammad Hamed Mozaffari & Lee, 2017)*. However, the primary motivation of those

studies is to track the probe in three-dimensional space (usually with 6 degrees of freedom (DOF)). In this study, we considered a simplifying assumption: The location of the ultrasound probe head and the face profile of the language learner are parallel respect to the camera lens during each language training session (see Figure 3-1.b and Figure 5-9). Under this assumption, tracking of the probe only requires the calculation of its location in a two-dimensional plane instead of three-dimensional space. For this reason, we selected two key points on the ultrasound probe, and the tracking problem was converted from three-dimensional space to two-dimensional space. It is noteworthy of mentioning that our system is capable of detecting un-alignments with a significant deviation from parallel orientations as a false detection in ProbeNet architecture.

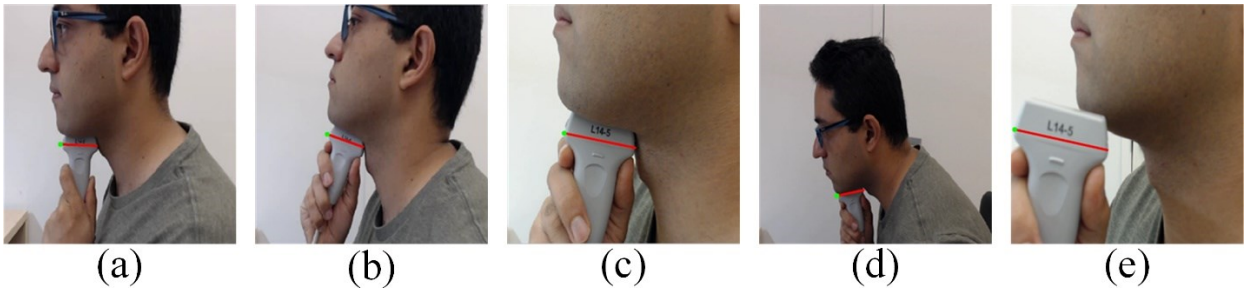



Figure 5-9. Tracking of the ultrasound probe in real-time. Two ends of the red lines show the place of two key points on the probe. Green point is considered for the sake of ultrasound frame transformations (it indicates the direction of the ultrasound probe).

Figure 5-10 illustrates the detailed architecture of the ProbeNet for ultrasound probe detection and tracking. It comprises of several standard convolutional layers followed by ReLU activation function and batch-normalization for more efficient training. In the last block, we used a dense layer with four neurons, which provides 2D positional information of the two key points as well as a dropout of 50 percent for better generalization over our dataset. During an L2 pronunciation training session, positional information of the two key points is used for tracking the ultrasound probe. These data are also used for transforming the current ultrasound frame on the language learner's face. ProbeNet is used for automatic tracking of the ultrasound probe in real-time. Transformations required for superimposing ultrasound data on face profile are calculated from the predicted output of the ProbeNet.

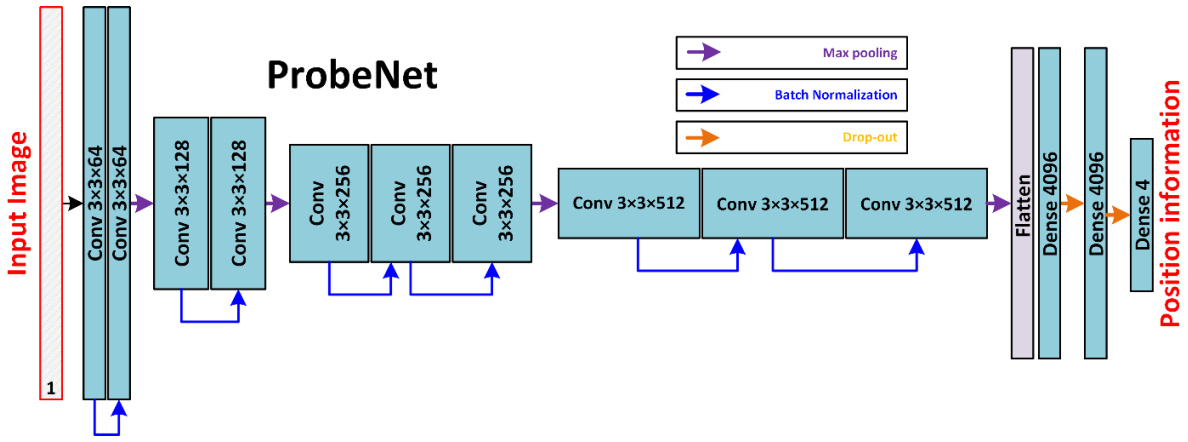



Figure 5-10. ProbeNet model architecture.

In order to train the ProbeNet for tracking the ultrasound probe, using our data augmentation toolbox (rotation, scaling, translation, and channel shift for images and the two key points), we created a dataset of 10000 images and their corresponding annotation information. Adam optimization algorithm, with the first and second momentum of 0.9 and 0.999, respectively, was used to optimized Mean Absolute Error (MAE) loss function during training and validation. A variable learning rate with an exponential decay rate of $10^{-6}$ and an initial value of 0.001 was selected for optimizing the model. We trained the ProbeNet model for ten epochs, each with 1000 iterations of mini batches of 10 images. Our experimental results revealed the robustness of the ProbeNet in probe tracking tasks. We got an average MAE of 0.0012 (with ±0.017 Standard deviation) for ten times running of the ProbeNet on our test dataset.

### 5.2.2 Using 3D Printable Stabilizer for Ultrasound Probe Tracking

In our previous experiments *(Hamed Mozaffari et al., 2019; M. Hamed Mozaffari et al., 2018)*, the user's head has flexibility without any stabilization due to the automatic face tracking. However, in this approach, different face profiles (different genders, skin characteristics, and ages) might provide a different result in each experiment, and ultrasound video is overlapped on the face with some non-accuracy. Also, in our freehand pronunciation training system, there is no restriction for ultrasound probe location and user's face. The freehand method cannot guarantee that the probe orientation is always in the mid-sagittal plane during a speech. Note that this is a restriction for linguistics quantitative studies, and for language training purposes, this limitation is not mandatory. In order to evaluate the accuracy of tracking the ultrasound probe, we designed UltraChin, which is a universal 3D printable device compatible with any ultrasound probes.

In addition to the aims aforementioned above, UltraChin is used for two other purposes in our system: I) As a reference marker for probe tracking module, II) For keeping the probe under the chin aligned with the mid-sagittal plane of L2 learner's face. UltraChin provides reference points for transformations, which are required to overlay ultrasound video frames on the user's face without any limitation for head movements. It is noteworthy to mention that UltraChin was created after several generations of designing, 3D printing, and testing on language learners (see Figure 5-11 for several generations of UltraChin). However, we utilized the last version of our UltraChin in experimental studies in the next Chapter. In order to make each part of the UltraChin, we used SolidWorks software. In the last generation (see Figure 5-12), we used natural materials in the process of 3D printing for skin sensitivity prevention due to the contact of human skin with plastic which was not considered in previous similar devices *(Scobbie et al., 2008; Spreafico et al., 2018)*. Furthermore, adding an extra part, users can attach other types of sensors, such as electromagnetic tracking sensors.

Unlike many previous helmets and stabilizer devices *(Derrick et al., 2018; Scobbie et al., 2008; Spreafico et al., 2018)* for ultrasound tongue imaging, UltraChin is a universal device capable of connecting to different types of ultrasound probe. It is fully printable by 3D printers, and our designs are publicly available for other researchers. One unique characteristic of UltraChin is that the ultrasound probe is held by the user, which makes the process of data acquisition more comfortable and even more accurate after training the user to keep the probe correctly. This ability enables users to utterance words with more comfort while the user sets the optimized pressure of the probe on the chin after several training sessions, results in better image quality and less slippage of the probe. Printable designs of UltraChin can be found freely on the internet[1]. The two markers are tracked automatically in real-time using our proposed deep convolutional network (ProbeNet). In this method, positions of the two key points on UltraChin provide us transformational information in each frame, comprises of probe orientation, location, and a reference for scaling of the ultrasound data.

[1] https://github.com/HamedMozaffari/UltraChinDesigns

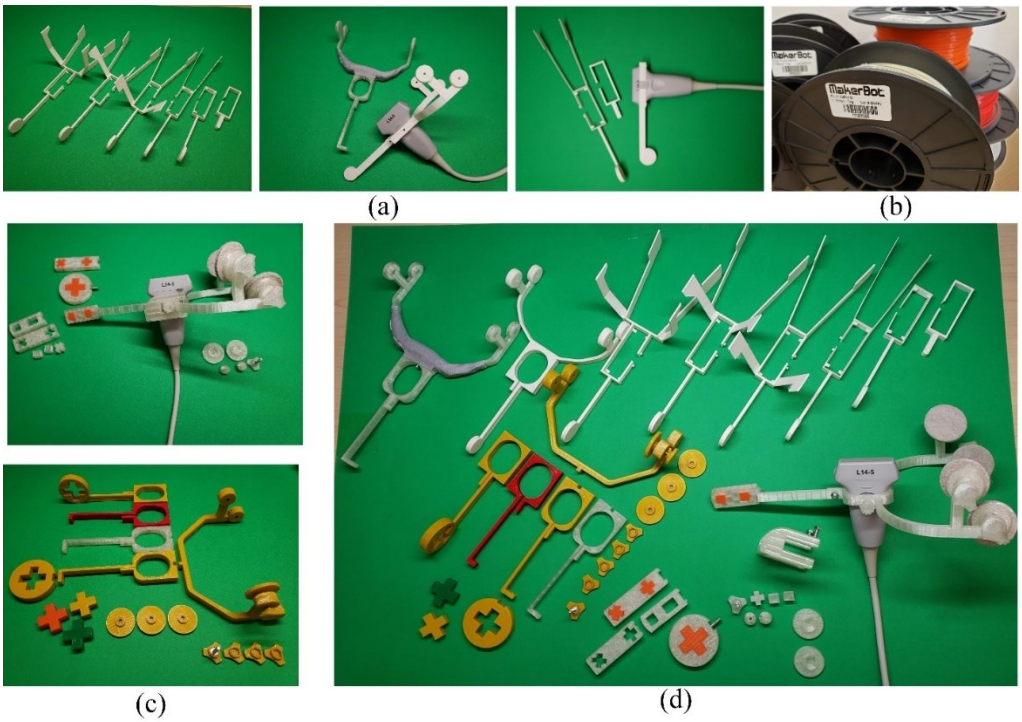



Figure 5-11. UltraChin: a) Integrated designs, b) Natural PLA materials for printing, c) Modular designs, d) Different versions and parts.


Figure 5-12. The last generation of UltraChin comprises of several modules. This version is universal and usable for different ultrasound probes as well as capable of being attached to other sensors.

# Chapter 6 Experimental Results and Comparison

As mentioned in the first chapters, new studies have revealed that visual feedback techniques, such as using ultrasound imaging, help individuals to acquire new foreign languages with higher efficiency. However, there are many difficulties in achieving a comprehensive language training system that shows students a real-time illustration of their tongue without the need for interpretation of noisy ultrasound images. This project aims to design, implement, and test different novel ideas to deploy a pronunciation training system as visual feedback by projecting the tongue movements on the correct place of the face view in real-time. The automatic tongue contour tracking method using state of the art deep learning approaches guides the user to focus on pronunciation training instead of localizing the tongue in ultrasound video.

Our implemented language pronunciation system is capable of assisting language learners in comprehending the pronunciation of difficult words and articulating them in their correct forms in real-time. Utilizing affordable, portable, and non-invasive ultrasound imaging device connected to a personal computer, achieving this goal change and shift the whole industry of foreign language training from traditional teaching techniques which concentrate on paper-based materials and listening of recorded audio to a novel interactive method. Benefiting from the real-time augmented reality technique, language learners learn how to pronounce foreign language words correctly without using any helmet or fixture to stable their head or ultrasound probe.

In this chapter, we explain the evaluation study of our language pronunciation training system. Because the main contribution of our project is on tongue contour tracking, we first investigate this topic thoroughly, then the whole system is described afterward.

## 6.1 Automatic and Real-time Tongue Contour Tracking

### 6.1.1 Fast Performance by sU-NET

Because the tongue contour in ultrasound data is a bright gradient region, automatic tracking of the tongue contour can be accomplished using image segmentation methods. Then, the contour curve can be extracted as a post-processing step using curve fitting, skeletonizing, or keeping top pixels of segmented regions. In the last few years, it was shown that a fully convolutional neural network (FCN) *(Long et al., 2015)* is capable of being trained for supervised end-to-end tasks such as image semantic segmentation.

There has been a trade-off between the number of training data, which is crucial in biomedical problems *(Litjens et al., 2017; Ronneberger et al., 2015)* and the number of trainable parameters in a network that means more computational cost and lower performance with limited facilities. An FCN model cannot provide sharp segmented results while there is only one up-sampling layer for mapping a relatively small encoder feature map to a large size output image. Thus, in order to address the first issue of small datasets as well as more accurate results, FCN was modified and extended to a new structure, similar to the deconvolutional network *(Noh et al., 2015)*, named U-NET *(Ronneberger et al., 2015)* such that it works with few training images, still with a precise segmentation result.

We adapted and modified the U-NET architecture for the task of automatic tongue contour tracking in real-time ultrasound video. To increase the performance of the U-NET model for this application, we truncated many repeated convolutional and deconvolutional layers. Our investigation indicates that many convolutional and deconvolutional layers in U-NET (with millions of parameters) practically improve the results slightly at the expense of substantial computational cost for both the training and testing process. As a consequence, our simplified version of U-NET (we named that as sU-NET) *(Hamed Mozaffari et al., 2019; M. Hamed Mozaffari et al., 2018)* is significantly fast. It can automatically delineate the tongue contour from ultrasound long video data in real-time applications. The primary motivation for proposing sU-NET is that ultrasound data of the tongue have a considerable correlated distribution where video frames are similar in terms of context and noise pattern amongst different datasets. Moreover, the location of the tongue change by time restricted to a limited set of postures.

Data augmentation is a powerful strategy for increasing the size of training data to allow a neural network to learn different realistic transformations and manipulations. In the case of tongue segmentation, although the tongue is flexible, it can only move in limited angles, and it can only deform in a limited length. Thus, data augmentation is limited to some specific transformations. We trained the sU-NET model on the SeeingSpeech dataset *(Lawson, E., Stuart-Smith, 2019)* consists of 6631 ultrasound tongue images (128×128), each with the same size corresponding annotated mask. The output image size of sU-NET decreased by valid paddings to (34×34) similar to the previous deep learning approaches *(Fabre et al., 2015; Fasel & Berry, 2010)*, where the output image size is 19×34. Therefore, annotated images are downsampled to the same size. Using small-sized images, we could compare our results with the previous methods where there is no

need to use a post-processing method, such as the tile mirroring strategy *(Ronneberger et al., 2015)*.

The SeeingSpeech database was downloaded as video files from *(Lawson, E., Stuart-Smith, 2019)*, and then each video was converted into frames. Tongue data were annotated using our annotation software, where a B-spline curve is fitted on annotated points by two experts. We trained sU-NET on the dataset with 80% training and 20% validation size for 50 epochs. The sU-NET network was deployed using Keras library *(Chollet & others, 2015)* with TensorFlow as its backend *(Abadi et al., 2016)*. For training of sU-NET, Adam optimization algorithm *(Kingma & Ba, 2014)* was used with a learning rate of 0.001, $\beta_1$ of 0.9, and $\beta_2$ of 0.999 and a schedule decay of 0.05 after each epoch. We trained and tested our model on a Windows PC with a CPU with 7 Cores at 3.4GHz and 16GB of memory.

Binary cross entropy (defined as Equation 6-1) was used as the loss function, which calculates the amount of error between the prediction and labeled data. To evaluate the proposed method performance, we also calculated the Dice coefficient (Equation 6-2) and Mean Absolute Error (Equation 6-3) whereas $p_i \in [0,1]$ is the *i-th* predicted instance from the last layer and $y_i \in [0,1]$ is the corresponding label for that *i-th* sample data.

$$L_{BCE} = \sum_i y_i \, log p_i + (1 - y_i) \log(1 - p_i) \qquad \text{Equation 6-1}$$

$$L_{Dice} = -\frac{2\sum_i p_i y_i}{\sum_i p_i + \sum_i y_i} \qquad \text{Equation 6-2}$$

$$L_{MAE} = \sum_i |y_i - p_i| \qquad \text{Equation 6-3}$$

We compared results of sU-NET with deep belief network (DBN) *(Fabre et al., 2015; Fasel & Berry, 2010),* which has already been used for the problem of ultrasound tongue segmentation. To do so, we used Mean Sum of Distance (MSD) *(Fasel & Berry, 2010; Karimi et al., 2019; M. Li et al., 2005)* as the Equation 6-4. MSD is applied on tongue contour extracted after skeletonization. This equation can provide an evaluation metric as the mean distance from pixels of a contour U to a contour V, even if the two curves do not share the same coordinates on the x-axis or do not have the same number of points *(Jaumard-Hakoun, Xu, & others., 2015)*. In order to extract contours with their corresponding coordinates, we employed the OpenCV morphological library for both predicted image and mask image. The former one made a contour set U of 2D points $(u_1, \cdots, u_n)$ and the later one made a contour set V of 2D points $(v_1, \cdots, v_m)$. Value of n and m can be set as equal for curves with similar length.

$$MSD(U,V) = \frac{1}{2n}\left(\sum_{i=1}^{n} \min_{j} |v_i - u_j| + \sum_{i=1}^{m} \min_{j} |u_i - v_j|\right) \qquad \text{Equation 6-4}$$

Few randomly selected instances of our proposed model using test set are shown in Figure 6-1. Our results reveal that the proposed method is outperformed previous methods for the tongue contour tracking and extraction in terms of accuracy. The average MSD value was 1.43 pixels for sU-NET, while it is 2.54 pixels for the DBN method *(Fasel & Berry, 2010)*. sU-NET could provide segmentation instance in 175 frames in 2.343 seconds, which equals 74.7 fps. The testing performance goes down to 29.8 fps when we add tongue contour extraction. Previous publications have not discussed speed, possibly due to the nature of semi-automatic or manual work. For the active contour tracking method mentioned in *(M. Li et al., 2005)*, the average MSD is 1.05 mm. It is essential to mention that the two human experts participating in the active contour tracking experiment produced two different annotation results having an average MSD of 0.73 mm *(M. Li et al., 2005)*, which may thus be reasonably considered the ultimate minimum MSD of training based automated methods. sU-NET outperformed other methods regarding MSD criteria, as shown in Table 6-1.

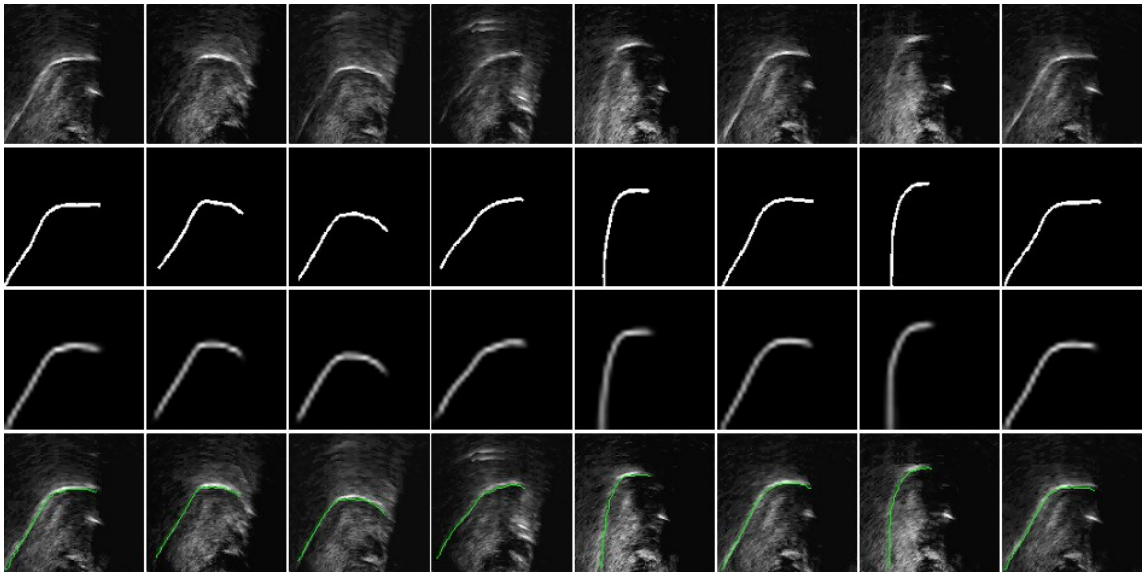


Figure 6-1. Results of applying sU-NET on video data. First row: Some randomly selected raw ultrasound frames (128×128), Second row: Corresponding masks for each frame in the first row (128×128), Third row: Predicted masks (34×34), Fourth row: Extracted contours using skeleton method on a resized version of third raw images (128×128) and then superimposed with the original frame.

Table 6-1. The comparison of our model with others in terms of average MSD. The sU-NET model shows better accuracy than other state-of-the-art methods on tongue dorsum extraction.

| Methods | sU-NET | DBN | Active Contour |
|---|---|---|---|
| Average MSD (mm) | **0.91** | 1.0 | 1.05 |

sU-NET tracking method can recognize the surface of the tongue in different positions and orientations due to the translation-invariant characteristics of the CNN models. In comparison with previous methods, sU-NET is the first fully-automatic and end-to-end deep learning model that predicts instances for the tongue contour tracking task in real-time data. It is noteworthy that due to the end-to-end training and testing schemes of the sU-NET, unlike previous approaches with two stages, there is no restriction for the number of frames. Therefore, sU-NET continuously tracks the tongue contour for extended video data.

In a separate training experiment, we trained sU-NET for more epochs to see the effect of the over-fitting problem. Table 6-2 shows the comparison for the Dice-coefficient validation error related to the number of epochs. The Dice-coefficient error declines to around 0.2 when the number of epochs is around 100 and then levels out. The DBN error remains around 0.4, even though the number of epochs increases.

Table 6-2. Comparison of our method and previous work in Dice-coefficient validation error. sU-NET outperforms DBN when the number of epochs reaches 50 and levels off at 100 epochs, at around half the error of the DBN system.

| Number of epochs | 5 | 50 | 100 | 250 |
|---|---|---|---|---|
| sU-NET | 0.446 | 0.243 | **0.212** | 0.212 |
| DBN *(Jaumard-Hakoun, Xu, Roussel-Ragot, et al., 2016)* | 0.41 | 0.38 | N/A | 0.4 |

sU-NET *(M. Hamed Mozaffari et al., 2018; Mohammad Hamed Mozaffari et al., 2019)* could predict better instances in comparison with previous methods for the problem of automatic tongue contour tracking. The real-time performance of sU-NET on CPU power provides a fast approach in this literature as an end-to-end architecture. However, the performance of sU-NET and previous deep learning techniques was evaluated on small-sized datasets. On the other hand, from the results of Table 6-2, under-fitting and overfitting for sU-NET and DBN can be seen clearly. Furthermore, for relatively small datasets such as tongue ultrasound data, the accuracy of these methods is not satisfying. Our experimental results also indicate that testing sU-NET on a different dataset while trained on another provides low-quality predictions, especially for big-sized images.

### 6.1.2 Fast and Accurate BowNet and wBowNet

To our knowledge, sU-NET is the first attempt to use a convolutional neural network for the problem of automatic tongue contour tracking. Evaluation results of sU-NET show its powerfulness in terms of speed with an outperforming accuracy in comparison to previous deep learning methods. Inspiring by U-NET *(Ronneberger et al., 2015)* and DeepLab v3 *(L.-C. Chen et al., 2017; Hamaguchi et al., 2018)* models, we designed two architectures from developing the sU-NET model. BowNet and wBowNet *(M. Hamed Mozaffari et al., 2019; M. Hamed Mozaffari & Lee, 2019b)* employ dilated and standard convolutions as well as skip connections. We aim to design a model with higher accuracy with the capability of testing on big sized images without compromising the speed performance of the sU-NET.

In order to train our BowNet and wBowNet models, first, we conducted an extensive random search hyperparameter tuning *(Bergstra et al., 2012)* for finding the optimum value of parameters in each network such as filter size (double for each consecutive layer starts from 16, 32, or 64), kernel size ($3 \times 3$ and $5 \times 5$), dilation factor (double for each consecutive layer starts from 1 or 2), the number of global iterations (iteration and epoch size), batch size (10, 20, and 50 depend on GPU memory), augmentation parameters (online and offline), dropout factor (0.5, 0.7, and variable), padding type (zero or valid), normalization layer (with and without), optimization methods (SGD and Adam *(Kingma & Ba, 2014)*), and the type of the activation layers (tanh and ReLU). We also tested several network configurations for the BowNet, whereas the number of layers was different in encoding-decoding forward paths (3, 4, and 5), and for the wBowNet, where the number of dilated convolution layers was different.

Our results from hyperparameter tuning revealed that, besides network architecture size, the learning rate has the most significant effect on the performance of each architecture in terms of accuracy. Testing fixed and scheduled decaying learning rates showed that the variable learning rate might provide better results, but it requires different initialization of decay factor and decay steps. Therefore, for the sake of a fair comparison, we only reported results using fixed learning rates. We evaluate four networks, sU-NET, sDeepLabV3, BowNet, and wBowNet, in this experiment. To have criteria, we also report results related to the original U-NET model. To investigate the performance of all networks in this experiment, we used a constant random seed value for

the initialization of all models and data augmentation. Table 6-3 presents selected learning rates for each proposed network model.

BowNet architectures were deployed using the publicly available TensorFlow framework on Keras API as the backend library *(Abadi et al., 2016; Chollet & others, 2015)*. We evaluate four networks, sU-NET, sDeepLabV3, BowNet, and wBowNet, in this experiment. To investigate the performance of all networks in this experiment, we used a constant random seed value for the initialization of all models and data augmentation.

Table 6-3. Fixed Learning-rate (LR) tuning using Best training and validation loss (BTL, BVL).

| | sU-NET | | sDeepLabV3 | | BowNet | | wBowNet | |
|---|---|---|---|---|---|---|---|---|
| LR | BTL | BVL | BTL | BVL | BTL | BVL | BTL | BVL |
| 0.005 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A |
| 0.0001 | 0.023 | 0.024 | 0.02 | 0.023 | 0.023 | 0.023 | 0.022 | 0.021 |
| 0.0003 | 0.021 | 0.023 | 0.018 | 0.021 | 0.021 | 0.022 | 0.021 | 0.023 |
| 0.0005 | **0.020** | **0.020** | **0.014** | **0.020** | **0.020** | **0.021** | 0.021 | **0.021** |
| 0.0007 | 0.019 | 0.022 | 0.017 | 0.020 | 0.020 | 0.021 | **0.020** | 0.022 |
| 0.0009 | 0.020 | 0.023 | 0.017 | 0.020 | 0.019 | 0.021 | 0.022 | 0.023 |
| 0.00001 | 0.026 | 0.028 | 0.026 | 0.029 | 0.027 | 0.027 | 0.026 | 0.031 |

For all the four deep convolutional networks, size of the input images, the number of iterations, mini-batch sizes, number of epochs were selected as 128 × 128, 3000, 10, 60, respectively. Adam optimization was utilized with $\beta_1$ of 0.9, $\beta_2$ of 0.999, and fixed learning rates from Table 6-3 for finding the optimum solution of a cross-entropy loss function. It takes less than one hour approximately for each time training of a model depend on the network size, using one NVIDIA 1080 GPU unit, which was installed on a Windows PC with Core i7, 4.2 GHz speed, and 32 GB of RAM. We also used the Google cloud virtual machine with a Tesla P100 GPU and 16GB of memory for acquiring training results faster.

One goal of designing BowNet models was to search for a model with robust performance with a significant generalization power on different datasets. For this reason, we evaluated deep network models using different scenarios (train and test on one dataset, train on one, and test on another dataset and opposite way). Two datasets in these scenarios comprise of ultrasound tongue images acquired with different ultrasound machines, one from OttawaSpeech (Dataset I) and another from the publicly available SeeingSpeech project (Dataset II) *(Lawson et al., 2015)*. From Chapter 4.1, both datasets were created after informed undersampling to have balanced data distributions. From

Figure 4-4, dataset I has higher variations than dataset II, and we consider them as heterogeneous and homogeneous datasets, respectively.

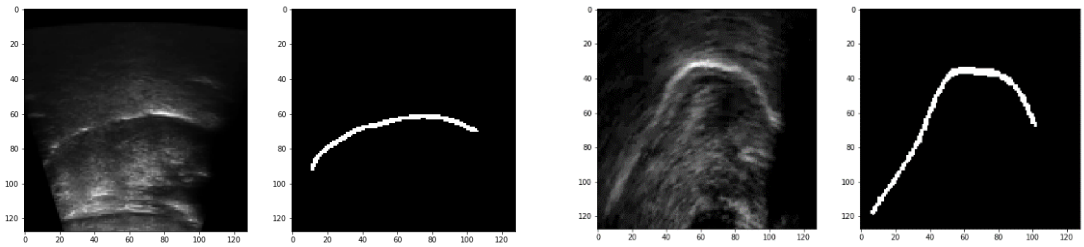
In all previous usage of machine learning methods for ultrasound tongue contour tracking, the ground truth labels are created as gray-scale images with values from zero to 255. From semantic segmentation literature, using gray-scale labels can be interpreted as segmentation of one image into 255 classes while the goal of tongue tracking applications is to track one class of tongue. Therefore, for the first time in this literature, we investigate the effect of using both binary and grayscale labeled datasets. True labels corresponding to each image were created by two experts manually using our customized annotation software. Users should only annotate several point markers on the edge of the tongue contour region, and then the contour curve will be created using the B-spline method automatically between point markers.

To create a binary label dataset, we thresholded the gray-scale annotated datasets using an optimized threshold value by try and error. We should repeat that as can be seen from Figure 6-3, after applying augmentation on truth labels, due to the interpolation for down- or up-sampling in the zooming process, they become gagged after binarization. Furthermore, our experiments show that the multiplication of the rotation matrix will slightly change intensity values due to the approximation in calculations. Moreover, some artifacts might be added to the data due to the online augmentation (see Figure 6-3 lower left column).

As Table 6-4 shows, we separated datasets into training, validation, and test sets. Each model was trained and validated on datasets separately using online augmentation, and then it was tested on both datasets separately. Because we used the same batch size, the number of iterations, the number of epochs, and the equal initial randomization, each model encountered with the same set of images during the training, validation, and testing procedure. Note that there is a corresponding annotated mask (ground truth) for each image in both datasets, which is augmented online, correspondingly.

Table 6-4. Specification of the dataset I and II for training using online augmentation.

| | Total number of images | Train/Validation (%80/10) | Test (%10) |
|---|---|---|---|
| Dataset I (OttawaSpeech) | 2058 | 1646/205 | 205 |
| Dataset II (SeeingSpeech) *(Lawson et al., 2015)* | 4016 | 3212/401 | 401 |



Dataset I (OttawaSpeech) Dataset II (SeeingSpeech) *(Lawson et al., 2015)*

Figure 6-2. Two sample images from Dataset I and II accompany with their corresponding binary truth labels.

Data augmentation can be accomplished in two ways online and offline. In the former method, images are augmented in mini-batches during training, while for the latter method, all images are augmented before training of the model. In this experiment, we assessed the impact of both online and offline data augmentation for the problem of tongue contour tracking in ultrasound images. Online augmentation on ultrasound data was deployed using the Keras augmentation library with the same random numbers for the training of each deep convolutional network. In total, 60 iterations $\times$ 10 mini-batches $\times$ 50 epochs = 30000 augmented images were used for the training and validation of each network model. We used image flipping (half of the dataset after augmentation), rotation (50-degree rotation in each side) and zooming (ratio of 0.5x to 1.5x) to mimic all the possible transformation which can happen in ultrasound tongue data. Some randomly selected augmented data are presented in Figure 6-3. Table 6-5 shows the specification of datasets I and II after data augmentation. We also created a new dataset using a combination of the two offline augmented datasets with 100000 images and with the same train/test ratio.

Table 6-5. Datasets information after informed undersampling and data augmentations. Each dataset has two versions of grayscale and binary labels.

| | # of images | Informed undersampling | Offline augmentation | Online Augmentation | Training/Validation (%90/%5) | Testing (5%) |
|---|---|---|---|---|---|---|
| Dataset I (OttawaSpeech) | 2058 | 2050 | 50000 | 30000 | 45000/2500 | 2500 |
| Dataset II (SeeingSpeech) | 4016 | 2050 | 50000 | 30000 | 45000/2500 | 2500 |

**Qualitative Analysis:** To illustrate the efficacy of our proposed segmentation methods qualitatively, some randomly selected instances from our proposed models on test sets of datasets I and II are presented in Figure 6-4. As can be seen clearly, wBowNet

could achieve better prediction results in terms of noise. Training and testing on the same dataset using online augmentation, all the models in this study generate instances with acceptable noise cancellation. In the case of training on one dataset and testing on another one, results show a worse prediction for both sDeepLabV3 and sU-NET. It is noteworthy to mention that the correct position of the tongue contour curve acquired from skeletonizing should be shifted toward the edge of the intensity gradient between black and white regions *(Stone, 2005)*.

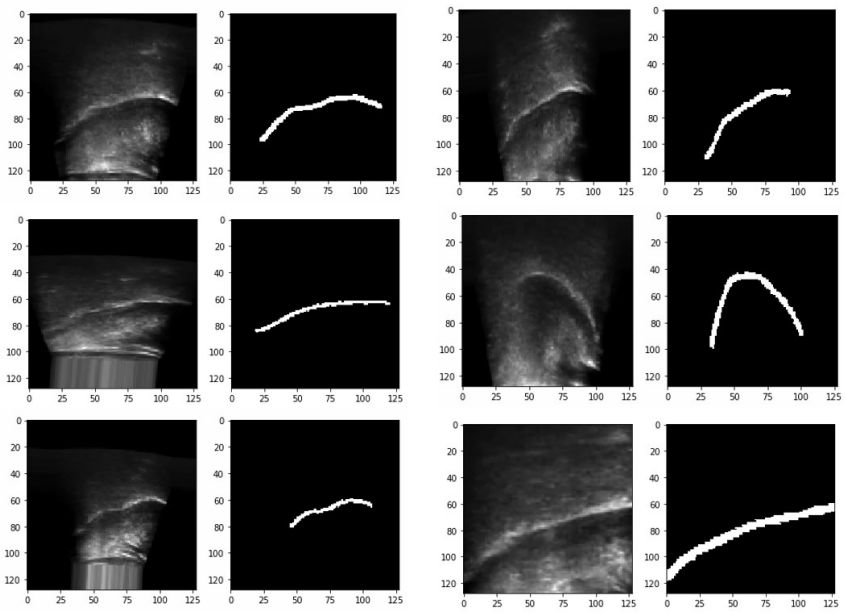


Figure 6-3. Some randomly selected images after online augmentation during the training process.

The segmentation results of the proposed models on datasets after enhancement and using offline augmentation is presented in Figure 6-5. Similar to the previous experiment, each model was trained on one dataset and tested on both datasets separately. To extract tongue contours, form the segmented region, we had to enhance the prediction map before applying the skeletonizing method for having a fair comparison. For this reason, first, the most significant object in the prediction images was selected by keeping the object with the maximum area (see the second rows of Figure 6-5), and then the skeletonization method was applied on the enhanced predicted map. The generated contours are compared with the skeletonized contour of the ground truth labels.

From Figure 6-5, we can see that the wBowNet could predict better feature maps with less noise and false prediction regions than other models. The last row of figures shows the difference between the prediction curve and the truth label curve. The faded area means a better correlation between the contour from the ground truth label and the

predicted map. By comparing the results of Figure 6-4 and Figure 6-5, we can see that BowNet architectures could predict instances with sharper edges when binary datasets are utilized instead of gray-scale labels.

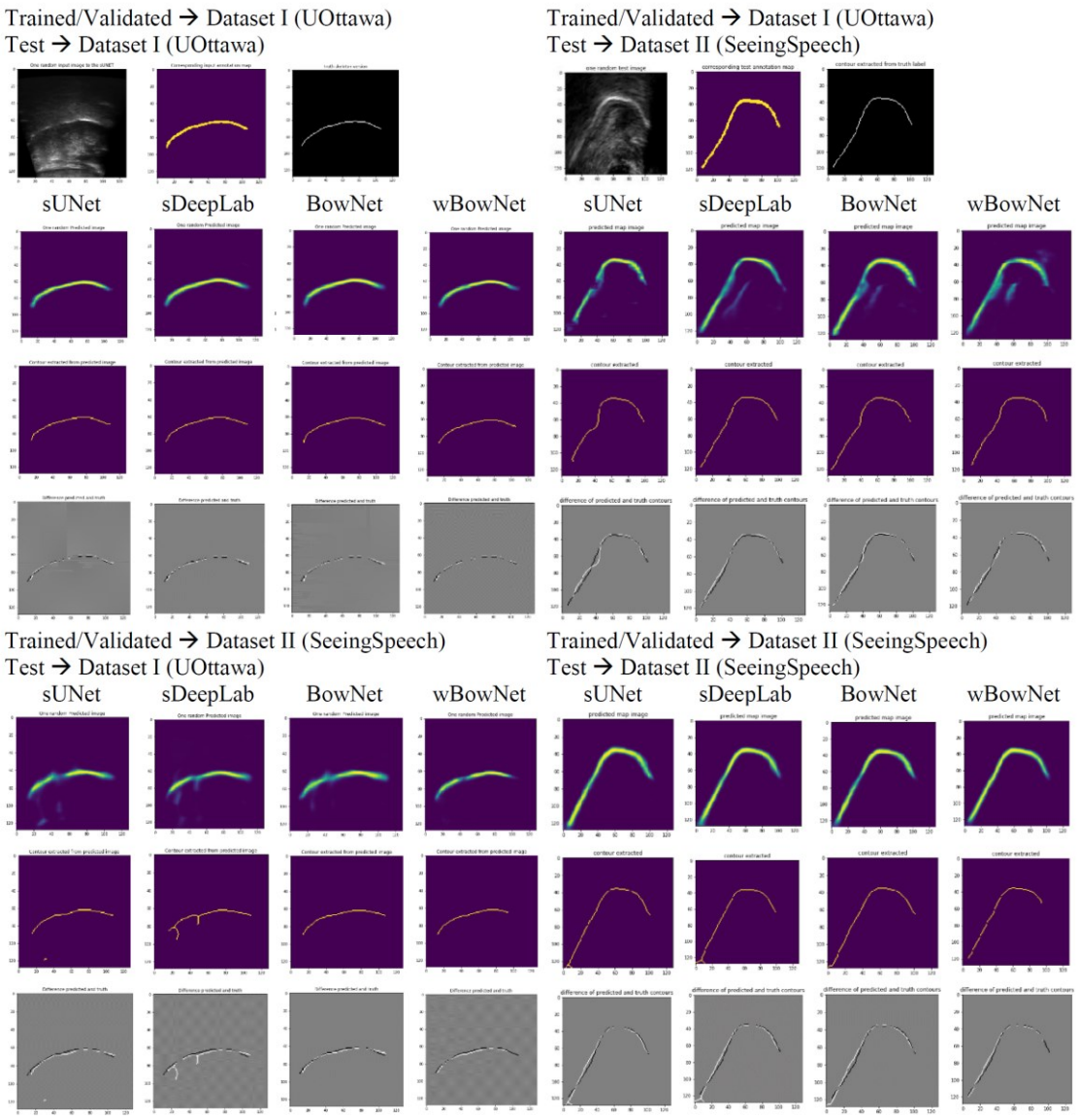



Figure 6-4. Sample instances from our four proposed models (online augmentation and grayscale). The first rows are prediction maps where yellow regions are a higher probability of correct prediction than other areas. Second rows are contour extracted from predicted feature maps, after the same binarization and skeletonization. The last rows are the difference between the contour extracted from the true label and the predicted label. Note that the original sample image, ground truth, and contour are illustrated on the top of the first row.

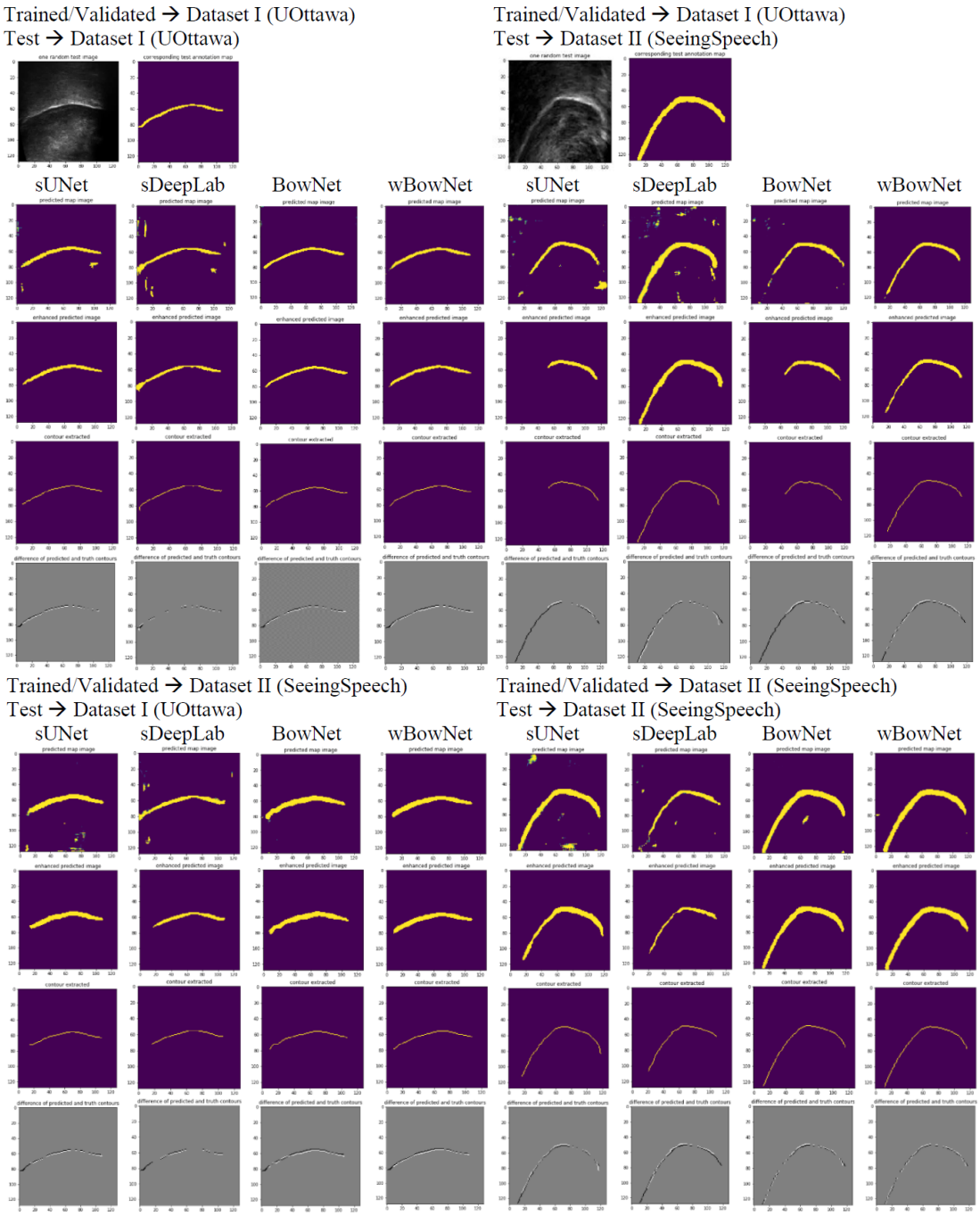



Figure 6-5. Sample instances of our proposed models (offline augmentation and binary). The first rows are prediction maps where yellow regions are a probability of correct prediction. Second rows are enhanced prediction maps using truncated smaller areas. Third rows are contour extracted from predicted feature maps, after binarization and skeletonization. The last rows are the difference between the contour extracted from ground true labels and predicted the label. Note that the original sample images and ground truth labels are illustrated on the top of the first row.

In a separate study, we trained and tested each model on a dataset that comprises of data from both datasets I and II using offline augmentation and enhancement. Then, each model tested on both datasets separately with the same configuration as before experiments. Figure 6-6 shows the consistency of our last conclusion about BowNet and wBowNet on the bigger dataset. As it can be seen clearly from the figure, both BowNet architectures could predict segmented images with less noise than other models. Difference images of tongue contours for sU-NET and sDeepLabV3 are analogous to the BowNet networks after enhancement of the feature maps, while those models could not achieve better predictions than BowNet and wBowNet.

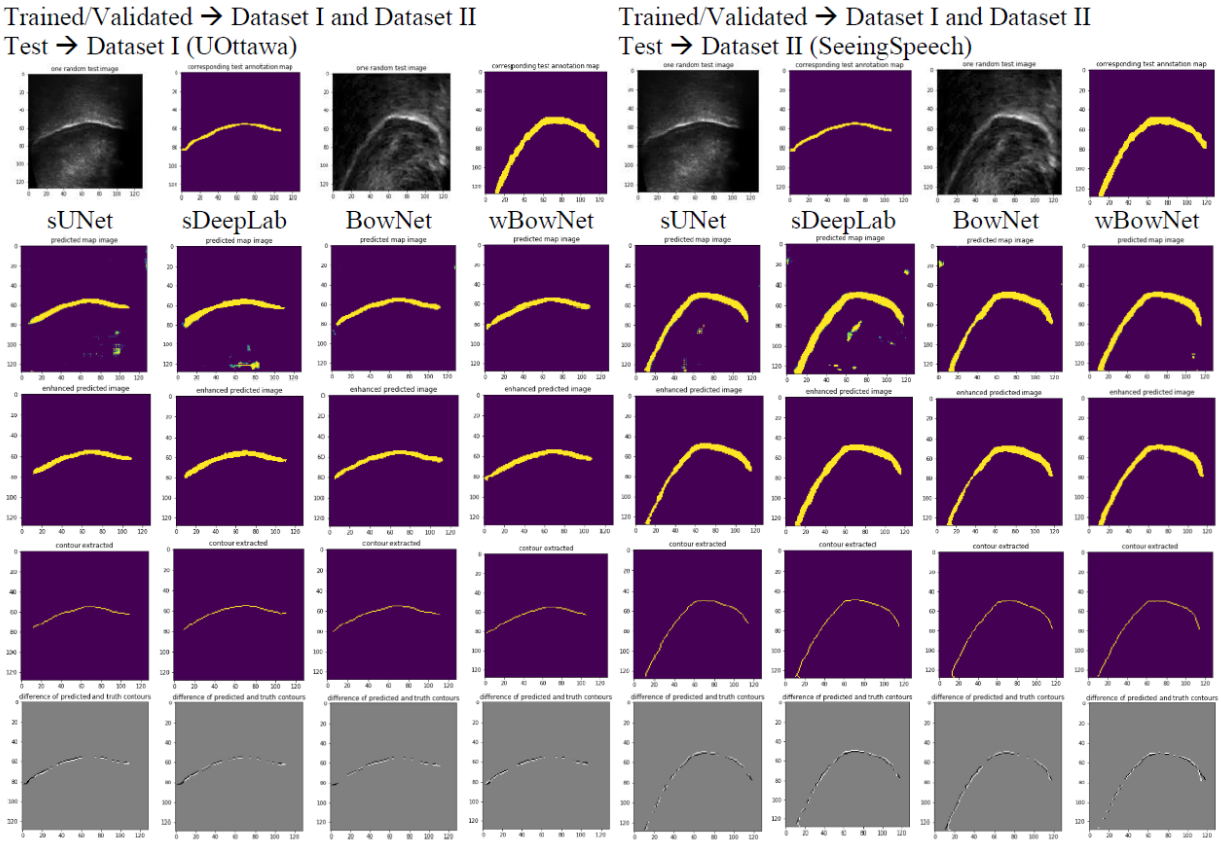



Figure 6-6. Sample test results on the combination of two enhanced datasets using offline augmentation. The first row is prediction maps where yellow regions are a higher probability of correct prediction than other areas. The second row is enhanced prediction maps. The third row is contour extracted from the feature predicted feature map after binarization and skeletonization. The last row is the difference between the contour extracted from the true label and the predicted label. The original sample image and ground truth of each dataset are illustrated on top of the first row.

**Quantitative Evaluation:** In order to quantitatively validate our proposed models, we followed the standard evaluation criteria for the field of ultrasound tongue segmentation *(Akgul et al., 1999; Fasel & Berry, 2010; Hahn-powell et al., 2014; Laporte & Ménard, 2018; Milletari et al., 2016; K. Xu, Gábor Csapó, et al., 2016)*. Contours extracted from prediction maps and ground truth labels are compared in terms of the mean sum of distance (MSD) as defined in Equation 6-4 *(Jaumard-Hakoun, Xu, Roussel-Ragot, et al., 2016)*. We carried out contour extraction using the skeleton method on binarized ground truth labels and predicted images employing morphological operators from the Scikit-learn library *(Martínková et al., 2004)*. The MSD criterion is sensitive to translation, as we saw in our experiments. For this reason, we did not shift the extracted contours to their correct positions for the sake of a fair comparison.

To evaluate the performance of the proposed methods, we also calculated and reported the value of the dice coefficient and binary cross-entropy loss between non-enhanced prediction maps and labeled data. These two criteria calculate the difference between the predicted and the ground truth labeled data, as defined in Equation 6-1 and Equation 6-2. Quantitative results are reported from the same experiments in the qualitative study with the same data, setups, and procedures. The results of testing each model using MSD criteria applying on the contours, extracted from enhanced prediction maps and the ground truth labels, are illustrated in Table 6-6. To convert MSD from pixel to millimeter, we calculated an approximated conversion of $1\ px = 0.638$ mm for both datasets.

Both wBowNet and BowNet could achieve MSD values around *0.04 mm* on average while the Deep Belief Network (DBN) model in a study by *(Fasel & Berry, 2010)* achieved *1.0 mm (1 px = 0.295 mm)*. For the active contour models mentioned in *(M. Li et al., 2005)*, the average MSD was 1.05 mm. It is essential to mention that again, the two human experts participating in active contour models experiment produced two different annotation results having an average MSD of 0.73 mm *(M. Li et al., 2005)*, which might thus be reasonably considered the ultimate approximate minimum MSD value-based human capability. In general, from the results of Table 6-6 and qualitative evaluation outcomes, BowNet models outperform other models while the training and testing datasets are from the same domain. In contrast, results of sU-NET and sDeepLabV3 are not reliable for datasets in the same or different domains.

Table 6-6. Results of testing each trained model using online augmentation on different datasets created. MSD is calculated and converted into millimeters with approximation depend on the ultrasound device resolution.

| Trained/Validated → Dataset I (OttawaSpeech) | | | | |
|---|---|---|---|---|
| Test → Dataset I (OttawaSpeech) | sU-NET | sDeepLabV3 | BowNet | wBowNet |
| Test loss | 0.02361 | 0.02711 | 0.02888 | **0.01896** |
| Test Dice | 0.75985 | 0.74899 | 0.74466 | **0.82629** |
| MSD (pixels) | 0.2496 | 0.2666 | 0.2819 | **0.2136** |
| MSD (mm) | 0.03744 | 0.0399 | 0.0422 | **0.0320** |
| Test → Dataset II (SeeingSpeech) | | | | |
| Test loss | 0.0829 | **0.06475** | 0.06964 | 0.06574 |
| Test Dice | 0.4985 | 0.52630 | **0.54718** | 0.52004 |
| MSD (pixels) | 0.4249 | **0.2408** | 0.3742 | 0.3588 |
| MSD (mm) | 0.0637 | **0.0361** | 0.0561 | 0.0538 |
| Trained/Validated → Dataset II (SeeingSpeech) | | | | |
| Test → Dataset II (SeeingSpeech) | | | | |
| Test loss | 0.04876 | 0.04540 | 0.04522 | **0.04367** |
| Test Dice | 0.71770 | 0.73029 | 0.71426 | **0.73680** |
| MSD (pixels) | 0.38279 | 0.39123 | 0.40143 | **0.26976** |
| MSD (mm) | 0.05741 | 0.05868 | 0.06021 | **0.04046** |
| Test → Dataset I (OttawaSpeech) | | | | |
| Test loss | 0.04400 | 0.04222 | **0.03948** | 0.06419 |
| Test Dice | 0.57989 | 0.57879 | **0.62013** | 0.39702 |
| MSD (pixels) | 0.26377 | 0.37512 | 0.2874 | **0.18037** |
| MSD (mm) | 0.03956 | 0.05626 | 0.04311 | **0.02705** |

In another experiment, we applied our proposed deep network models on enhanced datasets, which were created using offline augmentation. From Table 6-7, BowNet architectures have a better performance in terms of robustness where accuracy is considerable in all four scenarios. For instance, in two experiments, wBowNet achieved the best MSD values, and in the other two experiments, the MSD value for wBowNet has only about 0.002mm difference with the best architecture. Alternatively, sU-NET could reach the best MSD values in two experiments, but in two others, the difference with the best model was 0.01mm. In general, sU-NET predicts better instances when the OttawaSpeech dataset is for train and test, but with worse results on other datasets. This weak performance indicates the lower generalization ability of sU-NET and sDeepLabV3 in comparison with BowNet models. On the contrary, the performance of BowNet architectures is more steady for training and testing on the same or different domains.

Table 6-7. Results of each model on test sets of different datasets, created from offline augmentation.

| Trained/Validated → Dataset I (OttawaSpeech)<br>Test → Dataset I (OttawaSpeech) | sU-NET | sDeepLabV3 | BowNet | wBowNet |
|---|---|---|---|---|
| Test loss | **0.1447** | 0.2307 | 0.3242 | 0.1546 |
| Test Dice | **0.7617** | 0.6284 | 0.6243 | 0.7464 |
| MSD (pixels) | **0.2563** | 0.2619 | 0.2877 | 0.2665 |
| MSD (mm) | **0.0384** | 0.0392 | 0.0431 | 0.0399 |
| Trained/Validated → Dataset I (OttawaSpeech)<br>Test → Dataset II (SeeingSpeech) | sU-NET | sDeepLabV3 | BowNet | wBowNet |
| Test loss | 0.5427 | 0.5894 | **0.5378** | 0.5513 |
| Test Dice | 0.5815 | 0.5554 | **0.6216** | 0.5733 |
| MSD (pixels) | 0.4701 | 0.4639 | 0.4493 | **0.4159** |
| MSD (mm) | 0.0705 | 0.0695 | 0.0674 | **0.0623** |
| Trained/Validated → Dataset II (SeeingSpeech)<br>Test → Dataset II (SeeingSpeech) | sU-NET | sDeepLabV3 | BowNet | wBowNet |
| Test loss | 0.3255 | 0.4787 | 0.3796 | **0.1992** |
| Test Dice | 0.7514 | 0.6389 | 0.7313 | **0.8729** |
| MSD (pixels) | 0.4145 | 0.4024 | 0.4060 | **0.3300** |
| MSD (mm) | 0.0621 | 0.0603 | 0.0609 | **0.0494** |
| Trained/Validated → Dataset II (SeeingSpeech)<br>Test → Dataset I (OttawaSpeech) | sU-NET | sDeepLabV3 | BowNet | wBowNet |
| Test loss | 0.2524 | 0.3900 | **0.2227** | 0.2787 |
| Test Dice | 0.6795 | 0.5213 | **0.6947** | 0.6646 |
| MSD (pixels) | **0.2628** | 0.2659 | 0.2669 | 0.2761 |
| MSD (mm) | **0.0394** | 0.0399 | 0.0400 | 0.0414 |

In a separate experiment, all models were trained and tested on the combination of datasets (see Chapter 4.1). The results of Table 6-8 indicates that BowNet models are weak in terms of the Dice coefficient and Entropy Loss, but they are more accurate in terms of MSD values. From this experiment and previous results, we can conclude that the segmentation results of sU-NET and sDeepLabV3 have more noise and artifacts. For this reason, after applying similar enhancements and thresholding, BowNet models provide instances with sharper segmented regions. For this reason, the values of MSD, Dice, and Entropy loss should be considered together, and individual criteria do not provide valuable information about the performance of each model.

Table 6-8. Results of testing each model on a combination dataset from the dataset I and II where offline augmentation was used for the annotated labels.

| Test on OttawaSpeech | sU-NET | sDeepLabV3 | BowNet | wBowNet |
|---|---|---|---|---|
| Test loss | 0.19720 | **0.18100** | 0.29532 | 0.29040 |
| Test Dice | **0.73254** | 0.72778 | 0.65452 | 0.66786 |
| MSD (pixels) | 0.26962 | 0.26050 | 0.27034 | **0.25154** |
| MSD (mm) | 0.04040 | 0.03902 | 0.04050 | **0.03768** |
| Test on SeeingSpeech | sU-NET | sDeepLabV3 | BowNet | wBowNet |
| Test loss | **0.19530** | 0.21370 | 0.22566 | 0.23354 |
| Test Dice | **0.86066** | 0.83744 | 0.85592 | 0.85244 |
| MSD (pixels) | 0.38604 | 0.31790 | **0.29516** | 0.32756 |
| MSD (mm) | 0.05788 | 0.04766 | **0.04422** | 0.04910 |

As can be seen clearly in the last column of Figure 6-4 to Figure 6-6, both BowNet architectures could obtain outstanding prediction maps in comparison with other architectures. From the same figures, although BowNet models had significantly better outcomes in all qualitative experiments, the difference between contours was comparable for all the network models. For instance, sDeepLab has acceptable difference images, while prediction maps contain more noise than other models (see the last rows of Figure 6-5). Therefore, due to the post-processing stages on prediction maps, sU-NET and sDeepLabV3 obtained comparable results like BowNet models in the quantitative study, although they could not generate similar prediction maps. Therefore, the quantitative research in this work is support for the qualitative study, and the results of both evaluations should be considered together.

**Real-time Performance and Computation Cost:** the goal of this study is to propose a robust, fully-automatic, and at the same time with the capability of real-time performance. The real-time performance of a deep model depends on the network size, hyperparameter tuning, computational facilities, parameter initialization, per- or post data processing, data augmentation method, optimization method, and activation function. The first two items are more effective than others on the speed of the instantiation of each deep network model. Furthermore, the accuracy of deep learning methods is also highly related to the size of the training dataset and the complexity of the deep network model. Hence, there is always a trade-off between the number of training samples, which is a big issue in many applications such as in medical image analysis *(Litjens et al., 2017; S. Liu et al., 2019; Ronneberger et al., 2015)*, and the number of parameters in the network, which it requires more computing and memory units *(Badrinarayanan et al., 2015)*.

Increasing the number of datasets through data acquisition is not always the best and cheapest alternative. Data augmentation can help to alleviate this difficulty, but a bigger dataset needs a better network model in terms of generalization. In general, large state of the art deep network models *(L.-C. C. Chen et al., 2017; Noh et al., 2015; Ronneberger et al., 2015)* can be generalized on relatively small datasets such as the ultrasound tongue better with the expense of higher computational cost due to the number of parameters. For medical applications, both testing and training time are of great importance. For real-time applications such as ultrasound tongue contour tracking, a desired model should be fast in training and testing as well as accurate with an excellent generalization ability. Therefore, an optimized network benefits from new training strategies and efficient layer components could be an alternative for this favor *(S. Liu et al., 2019)*.

BowNet and wBowNet are capable of segmenting the tongue contour region from ultrasound images automatically result in similar or even better instances than larger architectures. Post-processing stages such as skeletonizing will slightly decrease the performance time while it became considerable when it followed by enhancements. From Table 6-9, sU-NET is the fastest method with double size of the BowNet. It can be seen in the table that the performance speed of each proposed method is in the real-time range. Figure 6-7 shows a sample of training and validating trends for the proposed deep models on datasets using online augmentation. Although the trends for all models are similar, slightly faster convergence can be seen for sU-NET and BowNet models.

Table 6-9 also shows the number of parameters and the memory which is needed to save those trainable parameters, calculated by the TensorFlow library (format of the memory is not specified, but it should be in bytes). As can be seen from the table, BowNet has the lowest number of parameters. wBowNet and sDeepLabV3 have almost double the number of parameters of BowNet. As an example, we calculated the number of parameters of the original U-NET as 31,042,369, with a similar method for our proposed architectures, which is 71 times bigger than BowNet and 39 times bigger than wBowNet.

Table 6-9. A comparison study of proposed models in terms of trainable parameters, the memory intake, and frame rate. Results are an average of 10 times run of models using test sets (using online or offline augmentation).

| | sU-NET | sDeepLabV3 | BowNet | wBowNet |
|---|---|---|---|---|
| Number of parameters | 948,833 | 785,889 | **434,785** | 786,657 |
| Memory | 3,795,332 | 3,143,556 | **1,739,140** | 3,146,628 |
| Framerate using GPU | 72 | 32 | 42 | 30 |

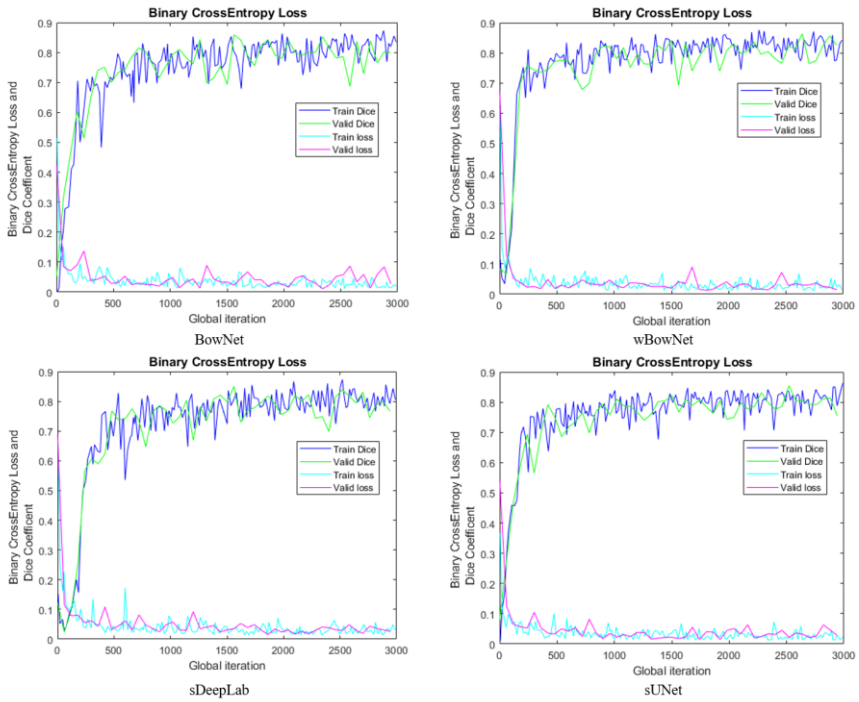



Figure 6-7. Training and validation trend for the proposed models. Note that for the sake of better presentation, results from every ten iterations is depicted on the horizontal axis.

### 6.1.3 Domain Adaptation for Tongue Contour Tracking

The outstanding achievements of convolutional neural networks on semantic segmentation task motivates us to propose an end-to-end model for automatic and real-time tracking of tongue contour in ultrasound video data. sU-NET inherited the majority of original model U-NET while with faster performance and few trainable parameters. However, the results of the sU-NET show a weak generalization ability and accuracy for testing that model on other datasets. BowNet architectures benefited from dilated convolution layers, improve the accuracy of sU-NET by keeping performance speed.

However, our experimental results revealed that none of our proposed models could be considered as a general model for applying on any ultrasound tongue dataset. In searching for developing previous deep learning models, we first investigate the usage of domain adaptation for the U-NET as a prevailing model *(Falk et al., 2019)* in medical image analysis literature to see the impact of transfer learning for the first time in ultrasound tongue contour tracking application. Transfer learning (we sometimes call domain adaptation) might provide a general solution for automatic, real-time tongue contour tracking and extraction, applicable to the majority of ultrasound tongue datasets.

Encoder-decoder models were successfully exploited for the semantic segmentation tasks *(L.-C. Chen et al., 2018)*. In this method, the encoder block of each deep learning model, pre-trained on a large dataset, performs as a feature extractor. For instance, the encoder of U-NET learns simple visual image features, especially in the first few layers, while the decoder aims to reconstruct the input-sized output prediction map from the complicated, abstract, and task-dependent features of the last layer of the encoder. For the problem of tongue contour tracking in ultrasound data, it is not apparent how much knowledge is preserved during the transfer learning process for domain adaptation.

In this experiment, the performance of the U-NET in different scenarios was analyzed to answer some fundamental questions in domain adaptation for this field. For detailed information about transfer learning, readers can refer to studies by *(Litjens et al., 2017; Pan & Yang, 2010)*. We examined how many layers from the decoder part should be fine-tuned to achieve the best segmentation accuracy in both the source and target domains at the same time (we called that a balanced point). Furthermore, the efficacy of dataset size in the target domain along with the skip operations and concatenations on the performance of the U-NET were explored on the problem of ultrasound tongue contour extraction. Using a trained deep network on one tongue ultrasound dataset from previous research cannot be generalized to other datasets. In other words, the results of each study are optimized for the personal dataset used for training in that project. In general, our primary motivation for this work is to investigate the impact of domain adaptation for a deep learning model as a base model, applicable to different ultrasound datasets with different distributions.

From Chapter 2.3.1, we saw that fully convolutional networks (FCNs) could be considered as dense classification networks (e.g., VGG-net *(Simonyan & Zisserman, 2015)*) with consecutive convolutional and pooling layers such that a fully convolutional layer substitutes the fully connected layer (e.g., SoftMax in the last layer). Similarly, DeconvNet is an FCN network with several deconvolutional layers in the up-sampling path. In U-NET *(Ronneberger et al., 2015)*, which is a DeconvNet architecture, feature maps (coarse contextual information) skips from each down-sampling layer to concatenate with deconvolutional layers for increasing the accuracy of output segmentation. The structural details of U-NET and DeconvNet are presented in Figure 6-8.

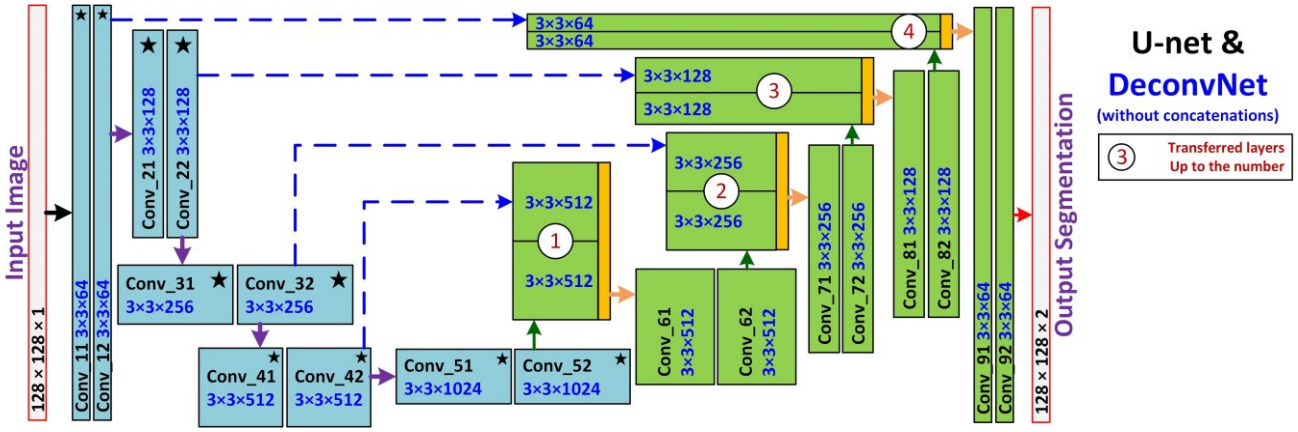



Figure 6-8. An overview of network architecture for U-NET and DeconvNet. Numbers in circles show several cases for finding the best model for transfer learning, where each number on a block indicates previous freezing blocks from training. For instance, number two means that we fine-tuned layers after that block. Previous layers are trained separately in advance. The image and filter size of each block is written on each box. Boxes, indicated by a star, show VGG-net *(Simonyan & Zisserman, 2015)* without the last fully connected layers. Note that the whole figure is a schematic of the U-NET model *(Ronneberger et al., 2015)* while excluding skip connections (dashed arrows) and concatenation layers, it is converted to DeconvNet model architecture *(Noh et al., 2015)*.

The DeconvNet comprises of 9 double convolutional layers of 3×3 filters with Rectified Linear Unit (ReLU) activation functions as non-linearity. Activations of all layers were normalized using batch normalization layers to speed up the convergence. In the down-sampling path, there are four max-pooling layers for the sake of translational invariance and saving memory by decreasing learnable parameters. In contrast, in the up-sampling path, there are four deconvolutional layers which retrieve the original spatial resolution. Finally, a fully convolutional layer at the end predicts the output feature map.

Network models were deployed using the publicly available Keras API with TensorFlow backend *(Abadi et al., 2016)* as the backend library. For the initialization of network parameters, randomly distributed values have been selected. Adam optimization was chosen with $\beta_1$ of 0.9 and $\beta_2$ of 0.999 for finding the optimum solution on a binary cross-entropy loss function. Each network model was trained using the mini-batch method employing one NVIDIA 1080 GPU unit, which was installed on a Windows PC with Core i7, 4.2 GHz speed, and 32 GB of RAM. Testing fixed and scheduled decaying learning rates showed that the variable learning rate might provide better results, but it requires different initialization of decay factor and decay steps. Therefore, for the sake of a fair comparison, we only reported results using fixed learning rates of 0.0005 (see *(M. Hamed Mozaffari & Lee, 2019b)* for more details about hyperparameter tuning procedure). To

alleviate the over-fitting problem, we regularized our networks by dropout rate of 0.5. Networks were trained for a maximum of 5000 iterations and a mini-batch size of 10.

We used two OttawaSpeech and SeeingSpeech datasets (see Chapter 4.1) as Dataset I and II. Models for from source to target domains $\tilde{f}_{ST}(.)$ were built from several scenarios with transferring the learned weights from $\tilde{f}_{S}(.)$ when we froze the encoder and some parts of the decoder. Specifically, in the scenario I, we transferred weights of the whole encoder block as well as portions of the DeconvNet decoder block as $\tilde{f}_{S}(.)$ which were learned on the dataset I $(D_S)$. Then we froze those blocks up to the i-th deconvolutional layer and fine-tuned the remaining $(4 - \text{i})$ deconvolutional layers using the Dataset II $(D_T)$ (see the circled numbers in Figure 6-8).

In scenario II, we investigated the opposite transferring case by switching source and target datasets to build the model $\tilde{f}_{TS}(.)$ to see the effect of reverse transfer. In similar scenarios, we also repeated the same experiments by considering the impact of skip operator and concatenation in U-NET to investigate the effect of transferring knowledge by injecting feature maps to the decoder from the encoder.

To evaluate each model, we investigated and compared different scenarios of tongue contour extraction as well. In each case, we first trained the whole DeconvNet and U-NET on the source domains (named base models), and then we directly applied them on two source and target domains to see the weakness of each model in terms of generalization from one domain to another. From Table 6-10, as it was anticipated, in both scenarios, base networks predicted better instances for their source domains than their target domains.

Outcomes of each scenario related to DeconvNet and U-NET have been presented in Table 6-10. From the table, we can assert that, on average fine-tuning, the whole decoder section is the best for achieving the best accuracy in the target domain. For instance, in the scenario I, in the case of the U-NET base model, it achieved a Dice coefficient (see (Hamed Mozaffari et al., 2019) for more details about loss functions) of 68.84% for the source domain and 46.64% for the target domain. At the same time, when the whole decoder fine-tuned, a better Dice coefficient of 63.06% was achieved in the target domain and 58.18% in the source domain. As it can be seen, by freezing more layer in the decoder section (conv7, conv8, and conv9) the difference between the Dice coefficient values in the source and target domains significantly increases. For the case of DeconvNet, this is not entirely true, and the difference decrease in higher layers.

Table 6-10 also indicate considerable result improvement in the scenario I for the U-NET compare to the DeconvNet due to the concatenation and skip operation. In

general, from the table, we can see that the decoder has a similar impact on the results as an encoder block, while fine-tuning more layers in the decoder section, higher accuracy results are achieved in one domain and negative learning in another domain.

Table 6-10. Quantitative results of each scenario. Reverse knowledge transferring can be seen in the two first columns for both models. Loss and Dice (in percentage) are cross-entropy and dice coefficient.

| Scenario I $DS \rightarrow DT$ | | DeconvNet | Transferred up to | | | | U-NET | Transferred up to | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Base | Encoder | conv 7 | conv 8 | conv 9 | Base | Encoder | conv 7 | conv 8 | conv 9 |
| Test $D_S$ | Loss | **0.2887** | 0.3221 | 0.3279 | 0.3742 | 0.4431 | **0.2269** | 0.3034 | 0.3203 | 0.3431 | 0.2467 |
| | Dice | **65.84** | 59.57 | 57.44 | 57.68 | 58.91 | **68.84** | 58.18 | 57.77 | 62.74 | 62.13 |
| Test $D_T$ | Loss | 0.4999 | 0.3573 | **0.3252** | 0.4100 | 0.4513 | 0.4805 | **0.3129** | 0.3600 | 0.3963 | 0.4627 |
| | Dice | 50.11 | **57.79** | 57.60 | 51.31 | 56.22 | 46.64 | **63.06** | 50.74 | 55.58 | 38.08 |
| Scenario II $DT \rightarrow DS$ | | DeconvNet | Transferred up to | | | | U-NET | Transferred up to | | | |
| | | Base | Encoder | conv 7 | conv 8 | conv 9 | Base | Encoder | conv 7 | conv 8 | conv 9 |
| Test $D_T$ | Loss | **0.3286** | 0.4423 | 0.4981 | 0.4856 | 0.4332 | **0.3736** | 0.5100 | 0.5777 | 0.5591 | 0.6015 |
| | Dice | **65.70** | 46.85 | 39.77 | 36.11 | 47.32 | **59.01** | 40.52 | 29.31 | 30.32 | 23.01 |
| Test $D_S$ | Loss | 0.4831 | **0.2571** | 0.2997 | 0.2986 | 0.2908 | 0.4253 | **0.2635** | 0.3363 | 0.3296 | 0.3270 |
| | Dice | 52.99 | **63.78** | 58.16 | 56.59 | 59.21 | 52.83 | **62.11** | 52.63 | 53.61 | 52.68 |

As it can be seen in previous studies such as *(Litjens et al., 2017)*, domain adaptation has not been made for encoder-decoder networks such as U-NET. A standard method of fine-tuning for domain adaptation is to use a pre-trained network (e.g., VGG16 trained on ImageNet dataset) following by a fine-tuning stage of the last layer, which is usually a fully connected/convolutional type. Our experimental results revealed that for achieving similar results in the case of ultrasound tongue image segmentation, instead of the last layer, the whole decoder should be fine-tuned to achieve better results in the target domain. We also observed that there is a balance point in encoder-decoder networks after training and refining that it can provide the best instances for both source and target domains.

To identify the sufficient size of the target dataset for transfer learning, in a separate experiment, we fine-tuned two transferred U-NET models (freeze the encoder and until conv 9) on three datasets with sizes of 100, 1000, and 10000 samples. We used the same network architecture and training procedure for this experiment. Figure 6-9 shows the difference values between dice coefficients and cross-entropy losses in source

and target domains for scenario I. From Figure 6-9 can be seen that more data samples enhance the performance of U-NET model in terms of accuracy in the case of transferring the whole encoder block while difference value for dice coefficient is not significantly different. At the same time, fine-tuning of the last convolutional layer of the U-NET will not improve its performance.

Figure 6-10 Illustrates the quantitative results of the scenario I for the U-NET model applied to a test instance. The U-NET (base model) was trained on the set of images from the source domain ($\tilde{f_S}$), achieving a test Dice coefficient of 67% and Binary cross-entropy loss of 0.38 while for the same model, the value of Dice and loss for the target domain was 48% and 0.49 without fine-tuning. It means that although the result of the target domain is not significant, the U-NET base model can still predict instances in both source and target domains. Nevertheless, in case of testing on a dataset with some frames contain rapid tongue movement along with noisy dorsum region (see Figure 6-9.c), the model fails in prediction for the target domain. This shows clearly that training one model, such as U-NET on one dataset, cannot be adequately generalized for other datasets from different ultrasound machines.

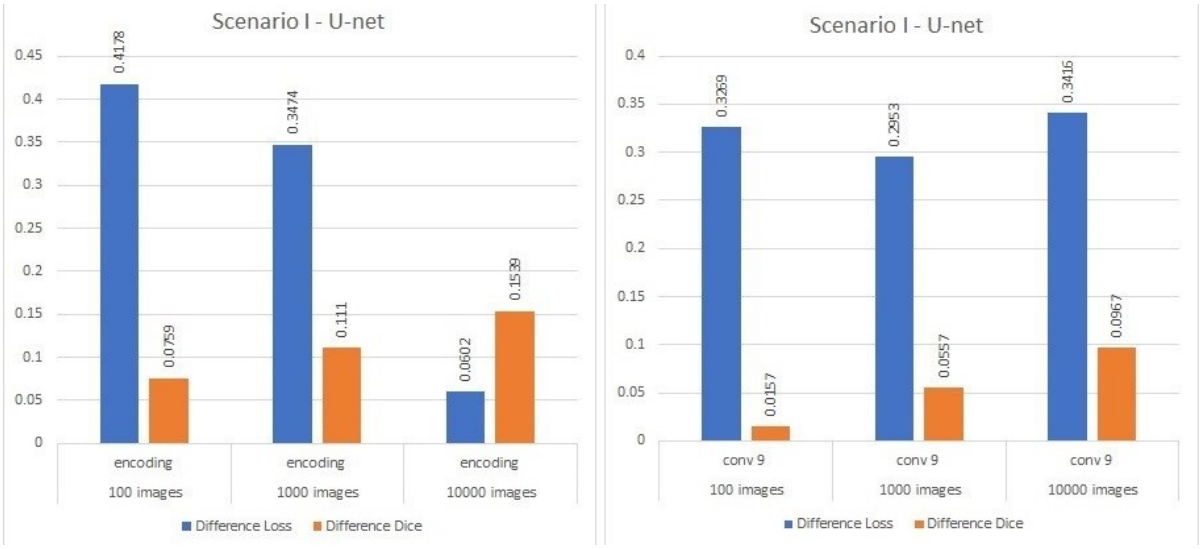



Figure 6-9. Effect of increasing dataset sizes on the accuracy of two transferred models in scenario I. Each column shows the difference of cross-entropy loss and dice coefficient between source and target domains. Left and right bars in each experiment are loss and dice values, respectively.

On the other hand, our experimental results revealed that fine-tuning of the whole decoder of the U-NET alleviates this problem significantly. For instance, dice score and loss values become 58% and 0.34 for the source domain and 56% and 0.39 for target

domains when the whole decoder fine-tuned on the target domain. Therefore, after fine-tuning, the results are similar in both domains. We observed a balance point for the number of refined layers considering both source and target domains. On the balance point, the model can achieve similarly acceptable results in both source and target domains, where the segmentation accuracy drops in total (see Figure 6-9.d). Besides algorithm design and engineering, dataset size and quality are two important factors for improving results. Fortunately, ultrasound tongue datasets have similar characteristics, such as analogous noises and artifacts. Finding a balance point for a general model such as U-NET over several ultrasound tongue datasets might provide a general pre-trained model network for the problem of tongue contour extraction without further fine-tuning for each target domain. However, the output accuracy is low for each dataset.

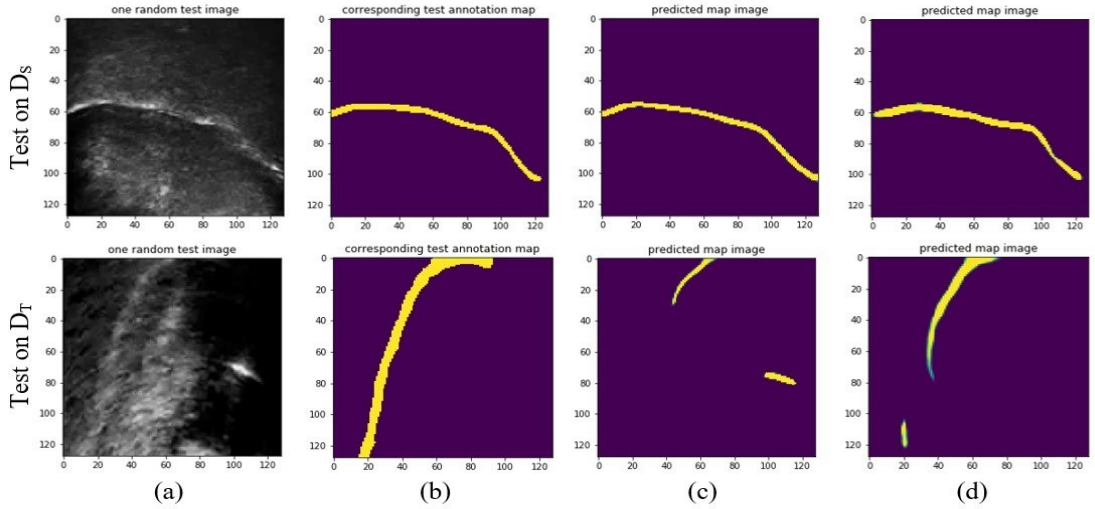



Figure 6-10. Prediction results of U-NET in the scenario I $D_S \rightarrow D_T$. (a) sample data, (b) corresponding ground truth labels, (c) prediction result of $\tilde{f}_S$ (U-NET base), (d) prediction result of $\tilde{f}_{ST}$ when the whole decoder was fine-tuned on $D_T$.

In transfer learning literature, researchers usually focus on finding $\tilde{f}_S$, which demonstrates a decent performance on a source domain $D_S$. Then they try domain adaptation from source to target $D_T$ task to find $\tilde{f}_{ST}$. This idea is an acceptable method among researchers in the machine learning field. However, a reliable and general approach is the one that can provide satisfactory results in the opposite path from target to source domains as well without compromising the reverse path. This could be more beneficial if the model works on more than one domain.

Similar characteristics of ultrasound tongue datasets enabled us to experiment with a different alternative for finding a general model capable of predicting instances among

different domains. Our experimental results showed that there is a balance point for U-NET model where it can provide reasonable predictions on both the source and the target domain ($\tilde{f}_{ST} \approx \tilde{f}_{TS}$). For instance, transferring the whole decoder of U-NET on the target domain, it provided Binary cross-entropy loss values of 0.3034 and 0.3129 for source and target test data, which are similar. Our real-time qualitative study over video frames also illustrates this accuracy balance between different domains.

Our qualitative research shows that domain adaptation can improve segmentation results for frames with significant noise and artifacts due to a better generalization of the network over different cross-domain samples. The impact of using skip operator, concatenation, and increasing dataset size in the target domain indicates a slight improvement in final results for U-NET in comparison to DeconvNet. In contrast with other research fields with large datasets, the size of a typical ultrasound tongue dataset is no more than $\sim 200K$ frames, and it makes more sense to fine-tune one model on several datasets to find the knowledge balance point as a general model for use in real-time applications on various ultrasound devices. Using lower learning rates in the target domains might increase the accuracy of the source and target domain segmentation further on the balance point. Further investigations over more than two datasets from different ultrasound machines are needed to generalize our findings in this experiment.

Unlike semantic segmentation literature with robust encoders pre-trained on large datasets, there is no powerful encoder block pre-trained on several ultrasound tongue datasets. In conclusion, utilizing domain adaptation to find a balance point between several ultrasound tongue datasets might be an alternative as a general method applicable for all datasets but with the expense of low accuracy.

### 6.1.4 Fast, Accurate, and Versatile IrisNet

In order to develop BowNet models for better generalization ability and to propose a model applicable for primary ultrasound tongue datasets, we designed IrisNet benefits from a novel convolutional module (RetinaConv) comprises of both dilated and standard convolutions. To show the generalization ability of IrisNet, we applied that for the problem of ultrasound tongue contour segmentation. There are plenty of ultrasound datasets private or publicly available for training deep learning models for automatic tongue contour tracking such as seeing speech project (SSP) *(Bliss, Bird, et al., 2018)*, University of Michigan (UM) *(J. Zhu et al., 2019)*, and University of British Columbia (UBC) *(M. B. Bernhardt et al., 2008)*, to name the most common once (see Table 4-1 for a complete list of datasets). However, none of them provides annotated data. We used

the OttawaSpeech tongue dataset contains 2085 annotated ultrasound images (see Chapter 4.1).

For the first time in the field of ultrasound tongue contour tracking, we followed the correct method of semantic segmentation by providing two binary labels for each image. There are two ground truth labels, one for background and another for the foreground (see Figure 6-11). Providing background label to a deep learning model assist the network in ignoring artifacts similar to the tongue in data. Furthermore, we increased the size of images in the dataset to 256×256, unlike previous machine learning techniques, which are trained on small-sized images.

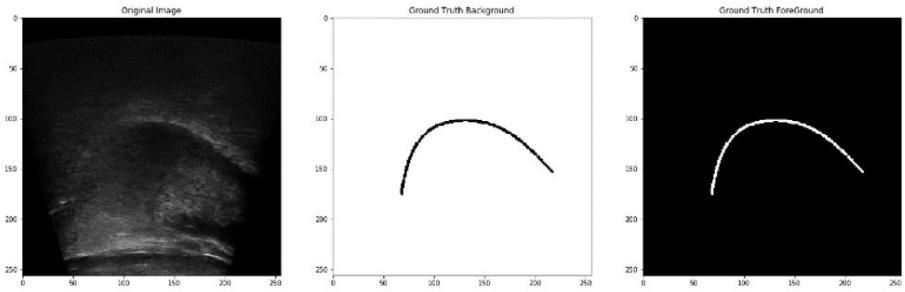



Figure 6-11. A sample frame from the OttawaSpeech dataset. The middle and right images are background and foreground truth labels (white and black are one and zero), respectively.

We divided the dataset into training, validation, and test sets by 80%, 10%, and 10% ratios, respectively. To increase dataset size, we used online data augmentation with realistic transformation factors, common in ultrasound datasets of the tongue, including horizontal flipping, rotation by a maximum range of 25 degrees, translation of 40 pixels shift in each direction and zooming from 0.5x to 1.5x scale. It is noteworthy that ground truth data in semantic segmentation literature are in one hot encoding format (binary images). Nevertheless, in the tongue contour tracking field, many previous studies used grey-scale labels for the training of their machine learning models *(Fasel & Berry, 2010; J. Zhu et al., 2019)*. In this experiment, we followed the method in *(M. Hamed Mozaffari et al., 2019)* for binarization of ground truth and online data annotation.

We trained IrisNet by Adam optimization *(Kingma & Ba, 2014)* method using its default parameters 0.9 and 0.999 for $\beta_1$ and $\beta_2$, respectively. We use a minibatch size of 20 images for 50 epochs. We test the sensitivity of the learning rate for IrisNet with different decay factors using a variable learning rate. The result can be better with different learning rates. For the sake of a fair comparison between models, we used a fixed learning rate of $10^{-3}$ for IrisNet, and for other models, we utilized their default values from each publication. For all models, we used random initialization for network weights. We

implemented and tested all models using one NVIDIA 1080 GPU unit, which was installed on a Windows PC with Core i7, 4.2 GHz speed, and 32 GB of RAM. We also used the Google CoLab with a Tesla K80 GPU and 25GB of memory for acquiring training results faster in parallel.

Dice loss and Binary Cross-Entropy loss *(M. Hamed Mozaffari et al., 2019)* were utilized for validation criteria. The training trend of IrisNet is presented in Figure 6-12. In both diagrams, satisfactory progress can be observed in terms of over-fitting and under-fitting. Less fluctuation was seen in our experiments for IrisNet in compare to other methods of this work. Note that Dice loss is (1 – Dice Coefficient), and it should be minimized.

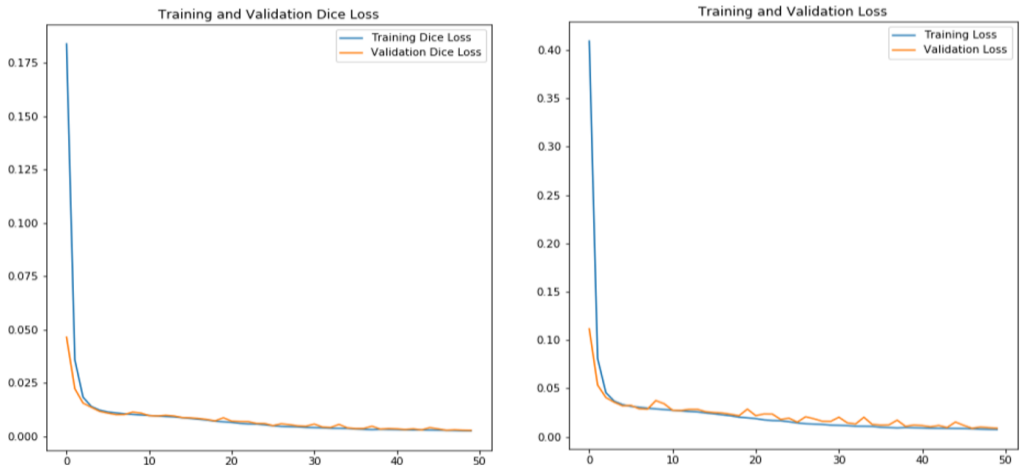



Figure 6-12. Training trend of IrisNet model on Ultrasound data. Left to right: Dice Loss and Binary Cross Entropy.

**Qualitative Study:** To assess IrisNet for the task of tongue contour tracking in ultrasound data, we train and compare results of recent and original similar deep learning models in the literature, including U-NET *(Ronneberger et al., 2015)*, FCN8 *(Long et al., 2015)*, BowNet and wBowNet *(M. Hamed Mozaffari et al., 2019)*. For each model, we used default values from each publication for their parameters. System setting, training procedure, annotated data, random seeds, and validation method, all were selected similarly for each model. We used checkpoint saving instead of stop criteria for the training of each model to keep the best-trained models.

To see which model predicted instances with less significant noise, we use the same threshold value for all models as 0.1. Table 6-11 presents values of intersection over union *(L.-C. C. Chen et al., 2017)* before and after thresholding instances as well as the average value for all samples in the test set (mIOU). IrisNet could predict instances with less noise

than other models. Note that the same test images have been used in both qualitative and quantitative evaluations.

Table 6-11. Results of each model in terms of Intersection Over Union (IOU) values before and after thresholding. The same threshold value and test images were used for all models (tIOU).

| Image | U-NET | | BowNet | | wBowNet | | IrisNet | | FCN8 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | IOU | tIOU | IOU | tIOU | IOU | tIOU | IOU | tIOU | IOU | tIOU |
| (1) | 87.7 | 38.9 | 84.1 | 36.8 | 87.7 | 40.7 | **87.8** | **40.8** | 87.7 | 32.8 |
| (2) | 91.7 | 42.5 | 89.2 | 38.4 | 91.7 | 42.8 | **91.7** | **42.9** | 91.7 | 37.1 |
| (3) | 94.6 | 48.4 | 94.3 | 54.3 | 94.6 | 49.5 | **94.6** | **53.2** | 94.6 | 39.7 |
| (4) | 88.9 | 39.5 | 87.1 | 33.0 | 88.9 | 36.5 | **88.9** | **36.6** | 88.9 | 35.9 |
| (5) | 83.4 | 39.2 | 81.1 | 42.1 | 83.4 | 40.4 | **83.5** | **45.3** | 83.4 | 38.7 |
| mIOU | 98.3 | 51.2 | 98.1 | 87.5 | 98.1 | 50.0 | **98.4** | **51.5** | 97.8 | 50.0 |

There are many linguistics methods for the evaluation of tongue contour tracking accuracy. However, we employed the standard techniques in the literature for testing machine learning models. To extract contours from predicted segmentation results, we apply a skeletonization method on ground truth labels and thresholded predictions (see Figure 6-13 and Figure 6-15) following the method in *(Karimi et al., 2019)*.

As can be seen from Figure 6-14 and Figure 6-15, IrisNet provides better instances in almost all cases in terms of disconnected regions and noisy segments in contrast to other similar techniques. For the same experiment, values of Mean Sum of Distances in terms of pixel and millimeters are reported in Table 6-12, while IrisNet could provide MSD values with better accuracy. Figure 6-14 presents our qualitative results of IrisNet for five randomly selected frames from the test set related to Table 6-11. IrisNet predicts less noise and false prediction in comparison to the other models. FCN8 predictions contain grey-scale squared shape because of using up-sampling instead of transpose convolution in the decoder section.

Table 6-12. Mean and standard deviation of Mean Sum of Distances (in pixels, 1 pixel ≈ 0.15mm) for 280 frame test datasets.

| Model | MSD (px) | MSD (mm) |
|---|---|---|
| U-NET | 4.15±0.72 | 0.62±0.39 |
| BowNet | 4.58±0.39 | 0.69±0.54 |
| wBowNet | 4.38±0.37 | 0.66±0.07 |
| IrisNet | **4.12±0.26** | **0.61±0.12** |
| FCN8 | 5.05±0.15 | 0.76±0.08 |

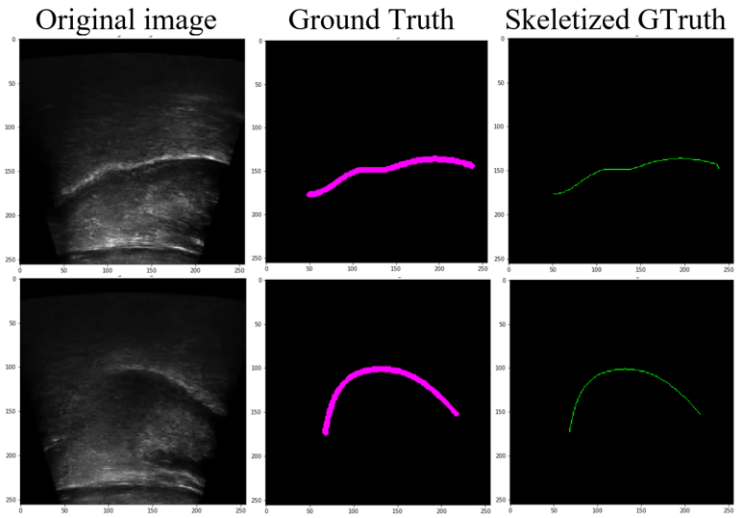



Figure 6-13. The curve (green) is skeletonized results determined from the ground truth label (pink).

We investigated the generalization ability of IrisNet for other datasets. For this reason, we selected random frames from each common publicly available dataset (EdgeTrak *(M. Li et al., 2005)*, UBC *(M. B. Bernhardt et al., 2008)*, SSP *(Bliss, Bird, et al., 2018)*, Ultrax *(Richmond & Renals, 2012)*, UM *(J. Zhu et al., 2019)*, UA *(J. J. Berry & James, 2010)*) and test IrisNet on each of those datasets. From Figure 6-16, although IrisNet had never seen any sample from testing datasets, it could predict segmentation results without any artifact of noise. Besides the generalization ability of IrisNet, one reason for this ability is that the test datasets from other institutes is relatively simple but with similar feature to the source OttawaSpeech dataset. Note that for the sake of representation, we warp test datasets in Figure 6-16. The quantitative results of the same study can be seen in Table 6-13. Except for the UA dataset, for almost all other datasets, IrisNet could predict better instances on average in terms of MSD.

Although IrisNet is superior to state-of-the-art deep learning models in ultrasound tongue segmentation literature, it has more trainable parameters than BowNet models and sU-NET, while its parameters are around one-fourth of the original U-NET model. On the other hand, it has faster performance in real-time applications due to the efficient implementation of the model. In Table 6-14, IrisNet has fewer parameters than the original U-NET, but it has similar performance speed using GPU.

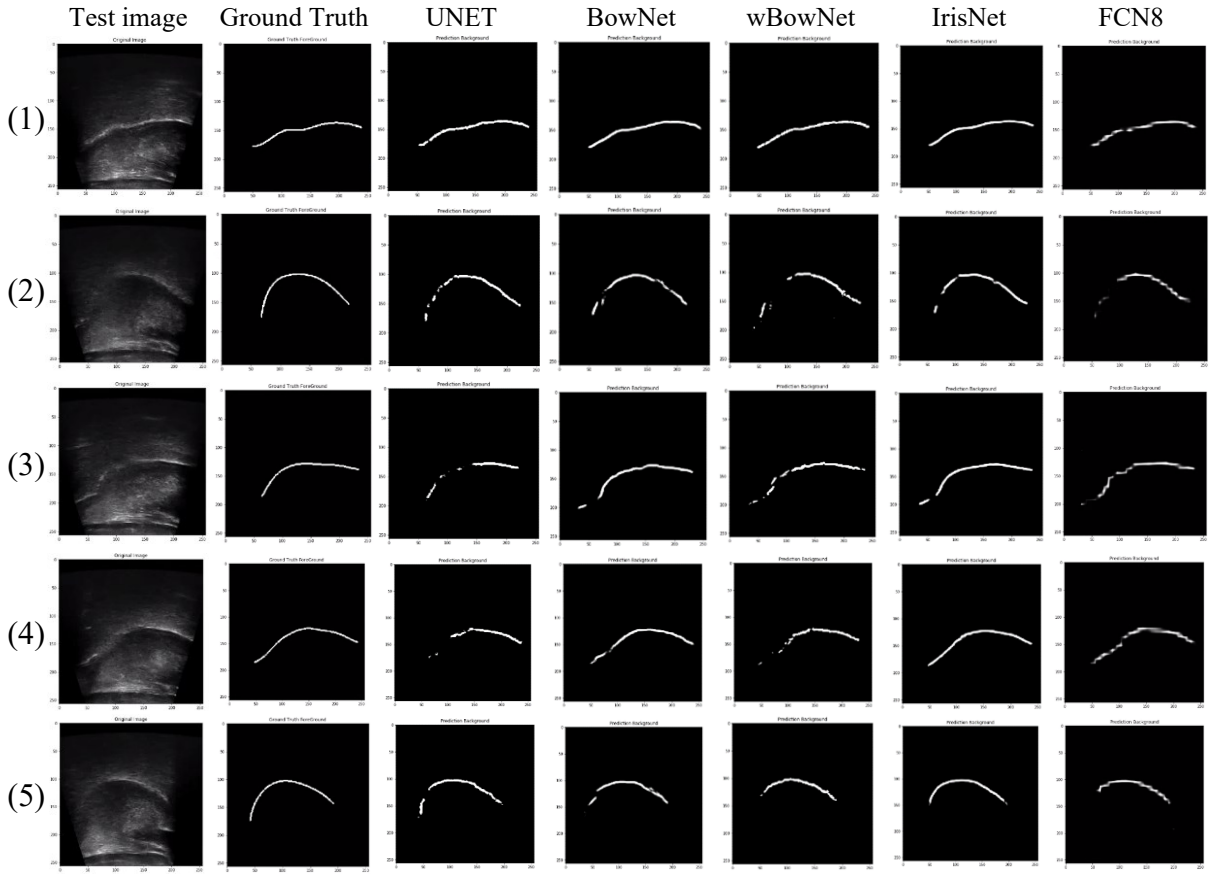



Figure 6-14. Sample results from the qualitative study. Each column shows the foreground predicted by each deep learning model. No pre- or post-processing has been applied for the dataset.

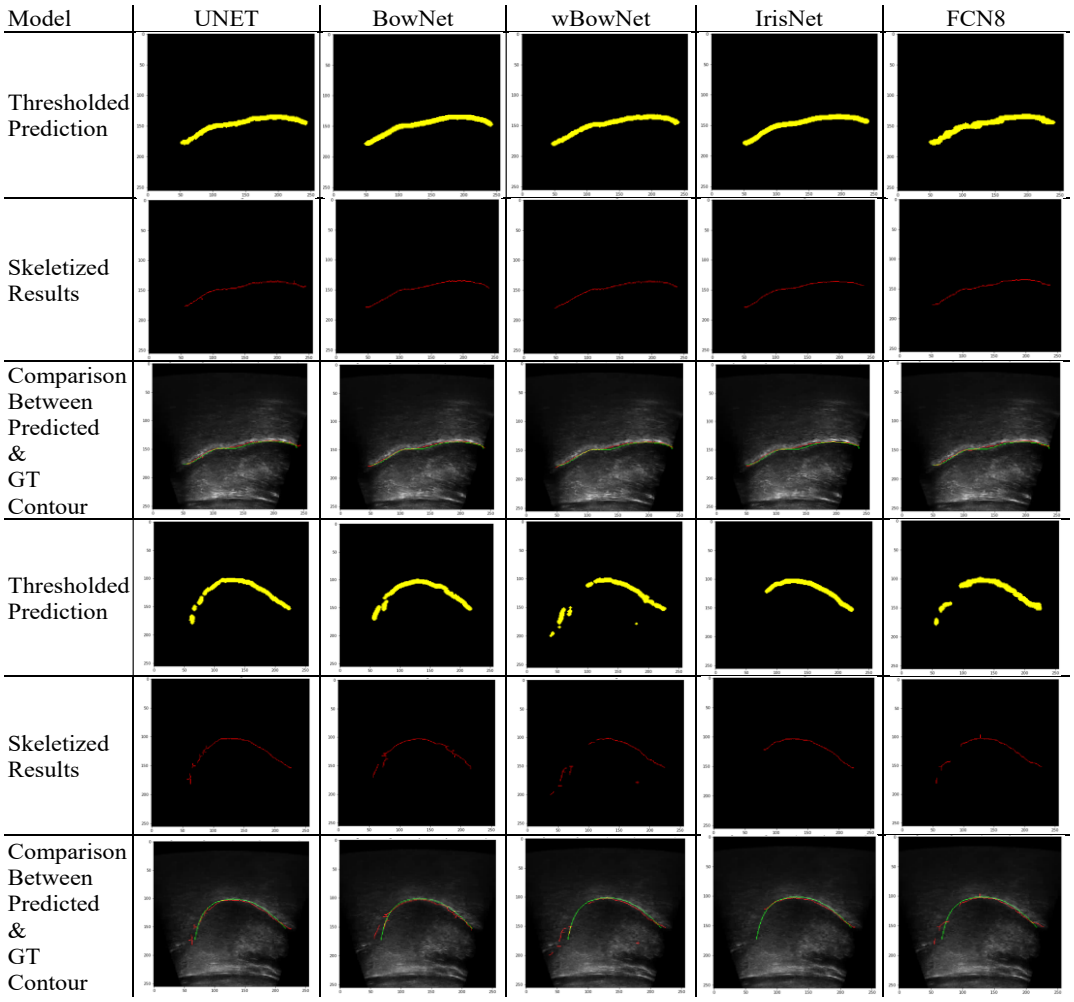



Figure 6-15. Comparing the contour extracted from segmentation results (red curves) with the contour obtained from ground truth labels (green curves) of each deep learning model for the two test samples of the previous figure.

Table 6-13. Mean and Standard Deviation of Mean Sum of Distances (in Pixels) for different test datasets comprises of 20 randomly selected frames.

| Model | UM *(J. Zhu et al., 2019)* | SSP *(Bliss, Bird, et al., 2018)* | UBC *(M. B. Bernhardt et al., 2008)* | UA *(J. J. Berry & James, 2010)* |
|---|---|---|---|---|
| U-NET | 5.27±0.81 | 6.31±0.25 | 5.42±0.73 | **6.83±0.53** |
| BowNet | 5.41±0.26 | 7.83±0.74 | 6.73±0.23 | 8.35±0.26 |
| wBowNet | 5.38±0.97 | 6.63±0.26 | 5.83±0.64 | 7.74±0.59 |
| IrisNet | **5.29±0.10** | **6.27±0.85** | **5.15±0.73** | 6.87±0.48 |
| FCN8 | 6.53±0.73 | 7.73±0.98 | 6.62±0.12 | 8.73±0.78 |

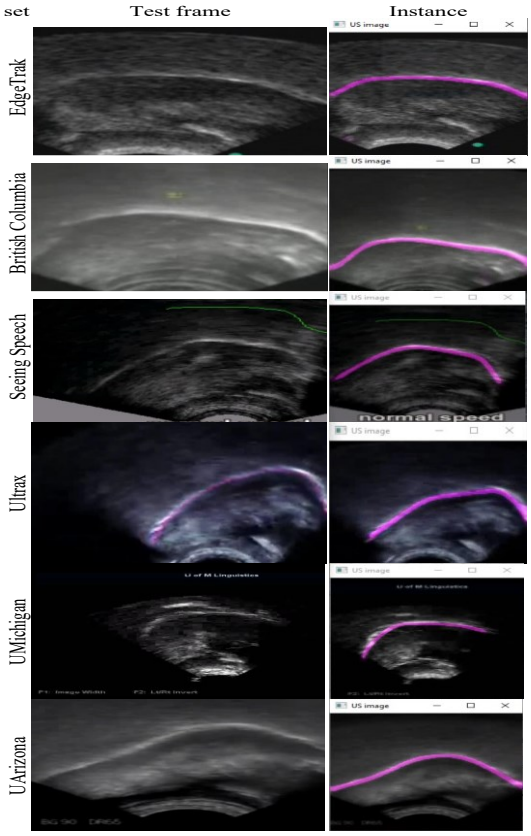



Figure 6-16. Testing IrisNet on sample data from standard ultrasound tongue image datasets.

Table 6-14. Performance speed (FRate in frames per second) and the number of trainable parameters (Params in millions). All models tested on GPU for ten times. The average and standard deviation values are reported for Frame Rate (f/s).

| | U-NET | sU-NET | BowNet | wBowNet | IrisNet | FCN8 |
|---|---|---|---|---|---|---|
| Params | 31.1m | 0.94m | **0.43m** | 0.79m | 5.93m | 134.2m |
| Frame Rate | 45±0.25 | **70+0.51** | 43±0.14 | 32±0.25 | 44±0.83 | 27±0.72 |

### 6.1.5 IrisNet for Tracking Points on Tongue Surface

As we discussed in previous chapters, machine learning methods have been successfully used for automatic tongue contour tracking of ultrasound image sequences. However, linguistic researchers usually use tongue contour for quantitative studies than the whole segmented region. For this reason, the procedure of tongue contour extraction has two-phase of image segmentation to delineated tongue surface region and post-processing contour extraction from the segmented region to find a one-pixel width curve. For instance, we used skeletonization and keeping the top pixel methods in our automatic methods. For real-time applications, with limited computational resources (for example, using smartphones), this approach cannot be a suitable solution.

As a novel idea and proposing future work in this field, we proposed a new method of tongue contour tracking using points. We trained only the whole encoder block of IrisNet (see Figure 5-7) for tracking a specific number of points on the tongue surface instead of the whole tongue contour region. The decoder block of IrisNet is substituted by three consecutive fully connected layers with 512, 256, and $k$ dimensional filters. The value of the $k$ is double size of the number of points, where each point has x and y coordinates as the output size of the last layer. A sigmoid function was used as the last layer activation, while %50 drop-out layers are used between each fully connected layer. For training of the proposed network, we used Adam optimization with its default hyperparameter values, and the learning rate is decreased exponentially from an initial value of 0.001 after each epoch. The proposed model was trained for 30 epochs of batches of 30 images. Mean Absolute Error is used as the objective function.

For creating datasets, we first manually annotated 300 cropped frames of the University of British Colombia dataset *(Adler-Bock et al., 2007)* (video related word "car") by 10 ground truth points instead of tongue segmentation annotated labels using our customized point annotating software. In a separate experiment, we automatically extracted several specific numbers of points using two strategies from annotated segmentation labels of the OttawaSpeech image dataset. In the first methods, we save the first high-intensity pixel for each column of the image in spaced distances. In the second approach, we first selected the top pixels of segmentation ground truth labels, and then we divided selected pixels into equally spaced points. Figure 6-17 shows randomly selected frames from testing the proposed model on the UBC dataset using a network output size of 10 points.

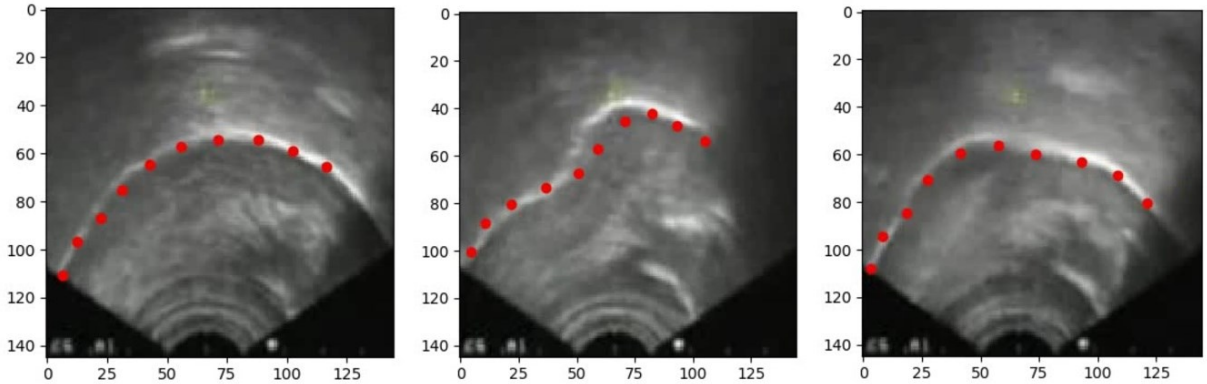



Figure 6-17. Randomly selected test frames from the UBC video dataset with a network output size of 10 points.

The same experiment on the OttawaSpeech dataset provides non-accurate instances. The reason is that the OttawaSpeech dataset is more complicated than the UBC dataset in terms of noise with a higher heterogeneous distribution. For this reason, we designed an online augmenter toolbox to increase the number of data for training of the proposed model. We used a randomly selected combination of values from 20 pixels translation in each side, 10-degree rotation angle in each side, and 0.5 to 2 times scaling ranges for data augmentation.

The first row of Figure 6-18 illustrates test instances when we trained our proposed model on a dataset created from saving 10 spaced points from all columns of the image automatically. In this experiment, we just trained the model on the dataset without augmentation. For this reason, we just save the best model before it overfitted on the dataset. In the second row of Figure 6-18, randomly selected test instances can be seen when we trained the proposed model using online augmentation. Data were automatically annotated by selecting equally spaced points from top pixels of segmentation ground truth labels.

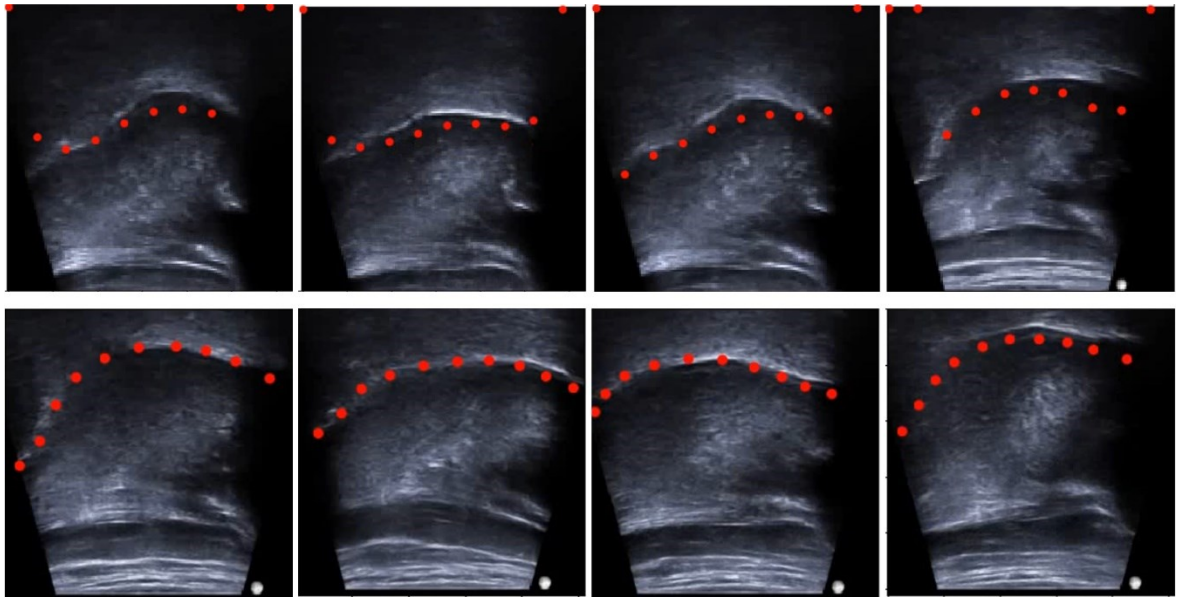


Figure 6-18. Randomly selected samples from testing the proposed model on the OttawaSpeech dataset. The first row is test instances before using data augmentation and uses the whole columns of the image. The second row are instances from the testing model when training dataset created by equal spaced points using data augmentation.

To find the best number of points as the output of the proposed model, we did an extensive experiment by training the proposed model on a different number of points in different annotated datasets using online augmentation. We made annotated datasets of

5, 10, 15, 20, 25, and 30 points, and trained the model separately using all these datasets. Test results revealed that the optimum number of points for the output of the proposed model is 10. From Figure 6-19, it can be seen that with increasing the number of output size, instances contain more non-accurate pixels. Although the current results are acceptable qualitatively, using a better objective function (such as penalizing non-accurate predicted points and considering information from neighborhood points during training), more accurate annotated samples and better selection of network parameters will improve results significantly.

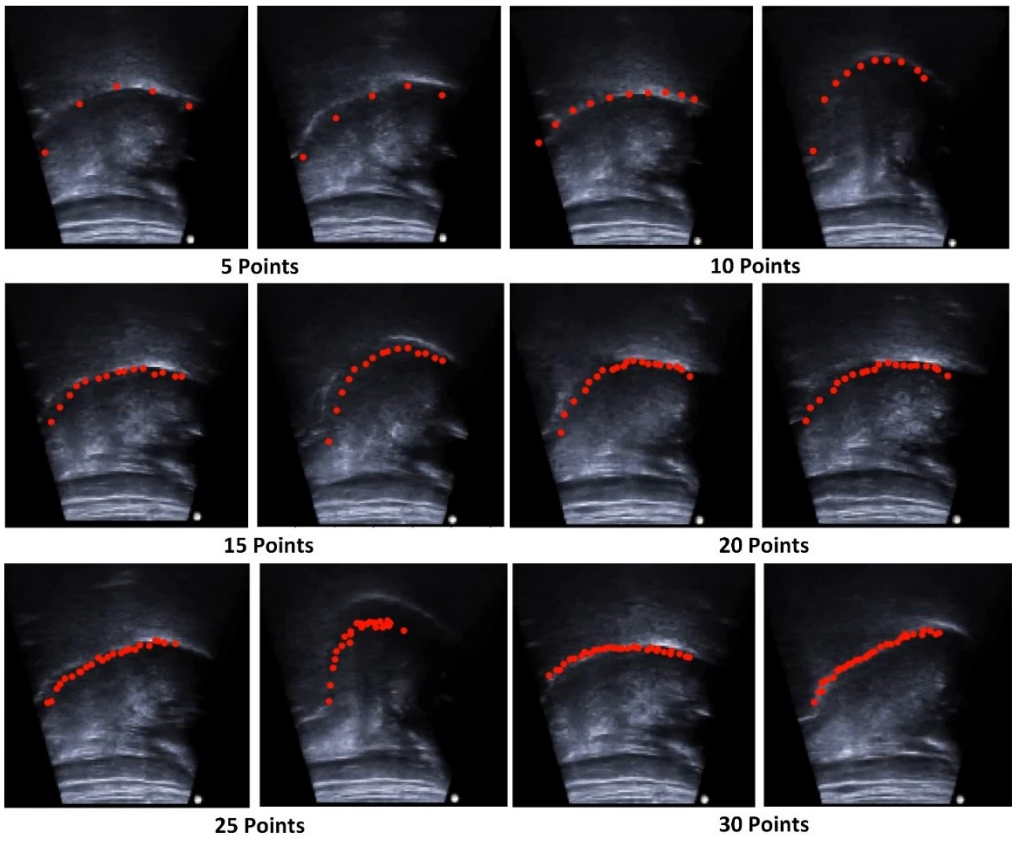



Figure 6-19. Instances from testing the proposed model with different output sizes.

## 6.2 Evaluation of Methods on Semantic Segmentation Benchmarks

We proposed a new framework using a new convolutional building block when there is no need for employing a pre-trained model as an encoder. In order to observe the powerfulness of IrisNet on other applications than ultrasound tongue datasets, we evaluated the IrisNet model on the Semantic Segmentation task followed by a comparison with state-of-the-art models in this field.

**Train without domain adaptation**: Besides factors such as recent advancements in computational machines (GPUs), large digital datasets, and efficient techniques, the major success of deep learning models on computer vision tasks owes to domain adaptation where weights of a pre-trained model *(He et al., 2016; Simonyan & Zisserman, 2015)* employed for fine-tuning of another model *(Tan et al., 2018)*. Nevertheless, using a model designed for image classification tasks, pre-trained on a large dataset, cannot be a reliable approach for fine-tuning of another model designed for semantic segmentation tasks. This issue becomes even more critical when the target domain is entirely different from the source domain *(S. Liu et al., 2019)* (e.g., Pre-trained VGG16 model *(Simonyan & Zisserman, 2015)* on ImageNet *(J. Deng et al., 2009)* for classification, fine-tuned for medical image segmentation).

Furthermore, publicly available encoders are trained for specific tasks, and there are usually restrictions for using their available pre-trained weights *(S. Liu et al., 2019)*. For instance, a non-modifiable network structure with a fixed-sized input image (e.g., PSPNet *(H. Zhao et al., 2017)*, DeepLabV3+ *(L.-C. Chen et al., 2018)*, and VGG16 *(Simonyan & Zisserman, 2015)* require squared sized images of 384×384, 513×513, and 224×224, respectively), forces researchers to manipulating (crop or interpolation) training data. An alternative technique is optimizing network architectures and improving their efficiency *(S. Liu et al., 2019)*. For example, variants of U-NET *(Ronneberger et al., 2015)* model are optimized, dominated, and applied in many medical image analysis tasks with outstanding results *(Falk et al., 2019)*, where suitable pre-trained models have not been available for most of these tasks. In *(Poudel et al., 2019)*, authors claim that for small networks, pre-trained models cannot boost performance.

Therefore, contrary to other studies focused on enhancing models for similar tasks and datasets, we intend to design a general deep learning model applicable for multiple datasets with distinct characteristics, usable for all researchers in different fields of science. Our proposed method is successful on scene parsing and semantic segmentation of different dataset types. One strength capacity of our model is the ability of training on different types of datasets with acceptable results without employing any pre-trained model. In this experiment, we evaluate IrisNet on two different datasets, including PASCAL VOC 2012 for semantic segmentation *(Everingham et al., 2015)* and CamVid for pedestrian and vehicle segmentation *(Brostow et al., 2008)*.

We implemented RetinaConv and IrisNet (see Figure 6-20) on the public platform TensorFlow *(Abadi et al., 2016)*. All models in this experiment were trained by Adam optimization method *(Kingma & Ba, 2014)* with first $\beta_1$ and second $\beta_2$ momentum of 0.9

and 0.999, respectively. Categorical cross-entropy loss is used for training, and in the last layer of all networks, we used "SoftMax" activation functions as the classifier. The learning rate value for all models was exponentially variable with iterations, initially set by 0.001 with the decay factor of $10^{-6}$. The performance might be slightly improved by increasing the epoch number, which is set by 100 for CamVid and 150 for PASCAL VOC. For data augmentation, we adopt random horizontal flipping, and random rescale between 0.5 to 1.5 as well as random shift with 10 percent in all directions for both datasets.

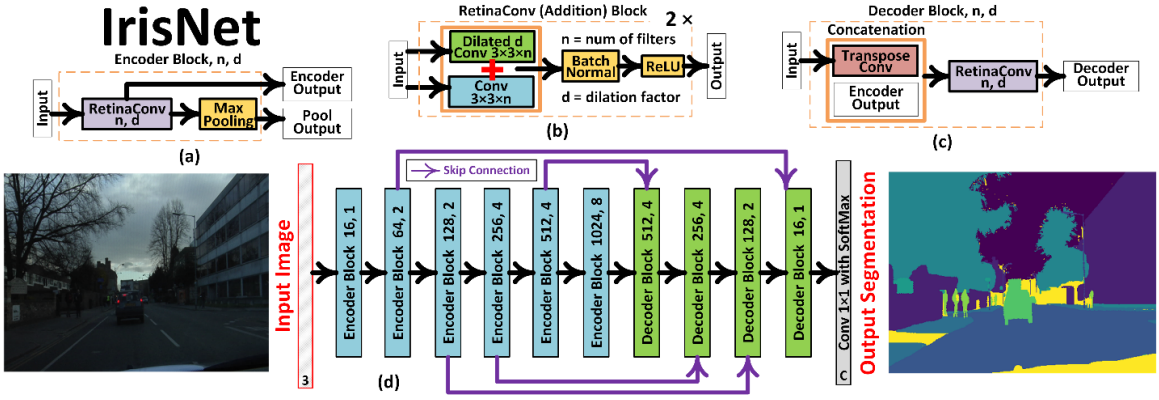



Figure 6-20. The network architecture of IrisNet.

Furthermore, for the CamVid dataset, we add a random Gaussian blur with a variance noise range of 0.2. Because of different dataset sizes, we cropped images in the online augmentation section to 320×320 for CamVid and 224×224 for PASCAL VOC. Following (Ioffe & Szegedy, 2015), we employed batch normalization instead of dropout layers between each layer. For network configurations and hyperparameters of models, we used default values from each publication or publicly available codes. For IrisNet, we followed the configuration of common encoder-decoders (Badrinarayanan et al., 2015; L.-C. Chen et al., 2017; Noh et al., 2015; Ronneberger et al., 2015) in the literature for a fair comparison between models. Activation function for IrisNet was ReLU, and due to the limited computational resources (GPU power), we selected the "batch-size" to 20 during training. For PASCAL VOC 2012, the ratio of train, validation, and test sets are 95%, 5%, and 5%. In the comparison study, we keep the best models by saving checkpoints during the training and validation stage, and for the testing stage, raw test images with their original sizes are fed to each network.

**PASCAL VOC 2012:** IrisNet works satisfyingly on scene parsing challenge of PASCAL VOC 2012 benchmark where the dataset has 20 objects categories and one background. Online augmentation of the PASCAL VOC dataset results in 7,863K, 438K,

and 438K image crops of 224×224 for training, validation, and testing. Table 6-15 shows the comparison results of IrisNet with several advanced methods on each benchmark. It is noteworthy to mention that we also attempted to test other best-performing models *(L.-C. Chen et al., 2018; T.-Y. Y. Lin et al., 2017; H. Zhao et al., 2017)* on this dataset. However, from our experimental results, none of these techniques provide considerable results (almost empty images) when random initialization is used instead of using pre-trained backbone models. One reason might be the inadequate training epochs or dead neurons during the training step because of a large number of parameters in these models (e.g., vanishing gradient or underfitting problems).

From Table 6-15, IrisNet outperforms other methods in terms of mean intersection over union (mIOU). For this reason, optimizing all sections of a network structure (even encoder part) is just as crucial as investigating other Influential aspects. Several instances of networks are illustrated in Figure 6-21. Although the results for all models are not considerable, IrisNet could predict instances with even more details, without employing the pre-trained weights of another model. For instance, the tail of the "cat" and ears/grass for "cow" have more details than other models.

Table 6-15. Performance of models in evaluation study on the PASCAL VOC 2012 test set in terms of IOU and mean IOU. The number of trainable parameters for each model is in millions.

| Model | Airplane | Bicycle | Bird | Boat | Bottle | Bus | Car | Cat | Chair | Cow | Table | Dog | Horse | Motorbike | Person | Plant | Sheep | Sofa | Train | TV | mIOU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BowNet | **62.7** | **41.5** | 16.3 | 17.6 | **53.8** | 35.6 | 41.9 | 67.2 | 13.8 | 67.8 | 12.9 | **85.4** | 63.2 | 49.1 | 38.4 | 51.7 | 67.2 | 23.8 | 27.6 | 21.9 | 51.2 |
| U-NET | 61.8 | 39.6 | 56.8 | 27.3 | 53.5 | 74.6 | 48.6 | **73.8** | 18.3 | **74.9** | 10.2 | 84.3 | **65.3** | 53.0 | 79.0 | **70.4** | **72.5** | **70.6** | 31.4 | 17.8 | 55.7 |
| FCN8 | 56.9 | 40.2 | 23.9 | 34.2 | 40.6 | 42.6 | 50.2 | 64.8 | 20.9 | 59.9 | 13.8 | 79.2 | 52.1 | 54.9 | 68.2 | 69.1 | 60.9 | 59.4 | 29.7 | 20.6 | 55.1 |
| LinkNet | 55.2 | 32.1 | 34.2 | **35.0** | 35.2 | 68.5 | 39.1 | 53.6 | 38.9 | 49.6 | 20.3 | 30.2 | 30.6 | 52.6 | 56.8 | 59.7 | 55.8 | 22.7 | 22.0 | 13.9 | 55.8 |
| IrisNet | 44.9 | 30.2 | **66.3** | 24.3 | 44.5 | **75.6** | **52.4** | 65.8 | **43.5** | 72.1 | **42.8** | 80.3 | 51.6 | **80.3** | **82.7** | 51.8 | 55.5 | 62.6 | **42.6** | **27.8** | **57.2** |
| FPN | 38.4 | 28.7 | 62.6 | 42.1 | 36.8 | 60.0 | 23.8 | 36.1 | 21.3 | 32.7 | 13.4 | 75.8 | 50.7 | 49.9 | 55.7 | 49.8 | 48.7 | 35.9 | 28.1 | 12.2 | 53.5 |

**A subset of CamVid:** CamVid dataset has 32 semantic classes for urban scene understanding. To compare each model on a more straightforward dataset, we employed a subset of CamVid contains three classes of "background", "car", and "pedestrian". In our evaluation study, there were 367, 101, and 233 annotated images for training, validation, and testing sets, while after online augmentation, training and validation sets were increased to 734K and 202K images, respectively. All models, except PSPNet (384×384), were trained with cropping sizes of 320×320.

We reported our assessment results of six networks on the CamVid dataset in three configurations (random initialization, initializing with pre-trained weights, and fine-tuning by freezing encoder parameters) while, except BowNet and IrisNet, the backbone network is VGG16 encoder network pre-trained on ImageNet dataset. For each configuration, we present three evaluation criteria mIOU, F1, and Categorical Cross-Entropy. From the table, IrisNet could predict better instances than other models, while random initialization was used for each model. Definitely, all models might achieve better results by optimizing all aspects of the experiment and training for more epochs. Some examples of this evaluation study are displayed in Figure 6-22. From the figure can be seen that although IrisNet is better in comparison with other models, it is weak in dealing with large objects in the scene (see also Figure 6-23).

Table 6-16. Quantitative results of the CamVid testing set. Methods without available pre-trained models are indicated by n/a. None means the network could not train on the dataset, without any prediction

| Method | Backbones ImageNet | Random initialization | | | Pre-trained initialization | | | Fine-tuning | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | mIOU | F1 | Loss | mIOU | F1 | Loss | mIOU | F1 | Loss |
| BowNet | n/a | 50.52 | 0.58 | 0.48 | n/a | n/a | n/a | n/a | n/a | n/a |
| U-NET | VGG16 | 35.80 | 0.38 | 0.87 | 37.58 | 0.41 | 0.86 | 0.40 | 0.43 | 0.81 |
| PSPNet | VGG16 | 32.18 | 0.33 | 0.93 | **46.50** | **0.54** | **0.65** | 54.80 | 0.63 | 0.50 |
| LinkNet | VGG16 | 38.83 | 0.44 | 0.80 | 39.19 | 0.44 | 0.80 | 65.06 | 0.73 | 0.36 |
| IrisNet | n/a | **55.77** | **0.61** | **0.45** | n/a | n/a | n/a | n/a | n/a | n/a |
| FPN | VGG16 | None | None | None | None | None | None | **68.44** | **0.76** | **0.32** |

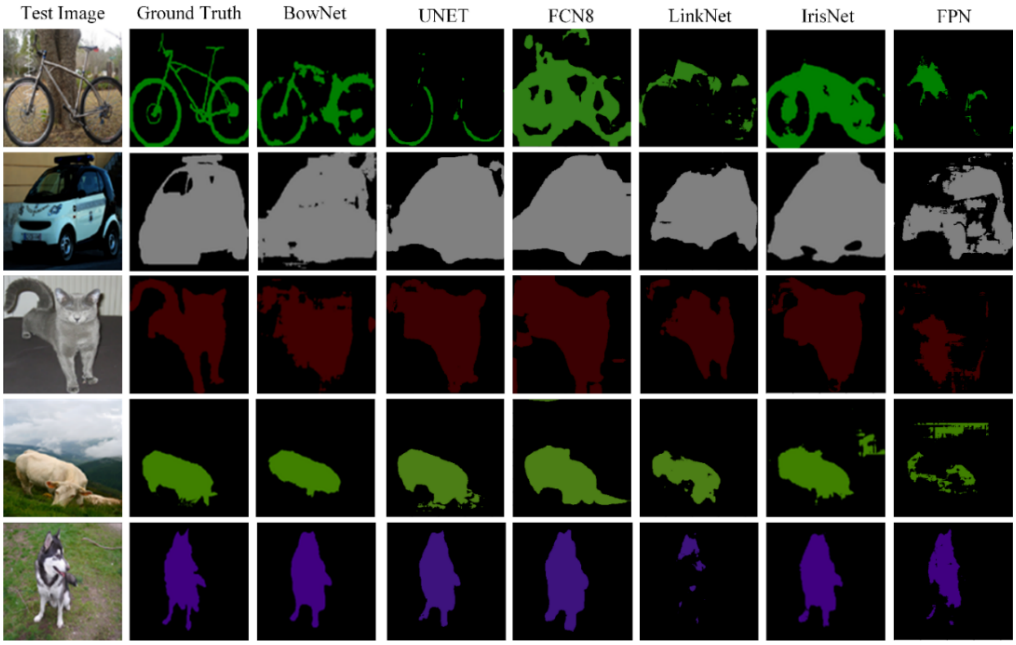



Figure 6-21. Results of each model in terms of per-class results on the PASCAL VOC 2012 *(Everingham & Winn, 2012)* testing set. All models are trained on the dataset without using pre-trained weights and initialized randomly.

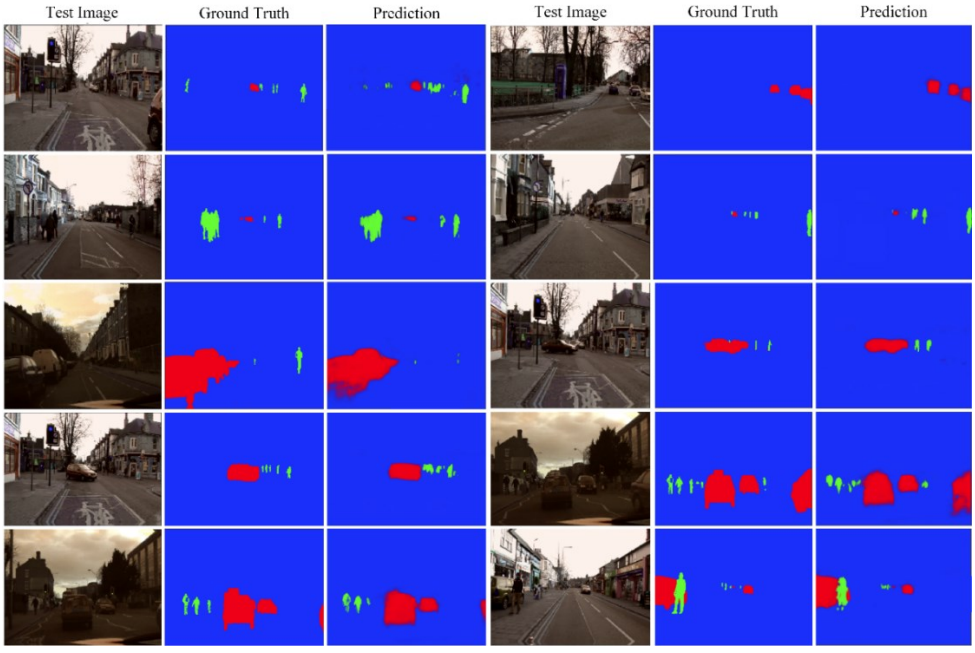



Figure 6-22. Results of the assessment of each model on the CamVid test set.

From this experiment, we can conclude that using deep learning models pre-trained on an abundant source dataset such as ImageNet will result in better instances after fine-

tuning on a target domain with the same context. However, fine-tuning is a difficult task with its disadvantages, such as negative training. Furthermore, a sizeable general dataset in ultrasound medical image analysis is not available yet. Therefore, IrisNet can be a promising alternative for the current small specific datasets as a general deep learning model. We investigated the powerfulness of IrisNet on the PASCAL VOC2012 dataset, and its performance is acceptable in comparison with fine-tuned models with a large embedded pre-trained model such as VGG16.

Although IrisNet and its proposed RetinaConv module outperform existing deep models in the literature, performance evaluation of the model on other datasets is a question. Furthermore, performance can be improved by using variable learning rates and increasing training dataset size by combining several sets. In the following sections, we investigate further the qualitative performance of IrisNet for a real-time application.

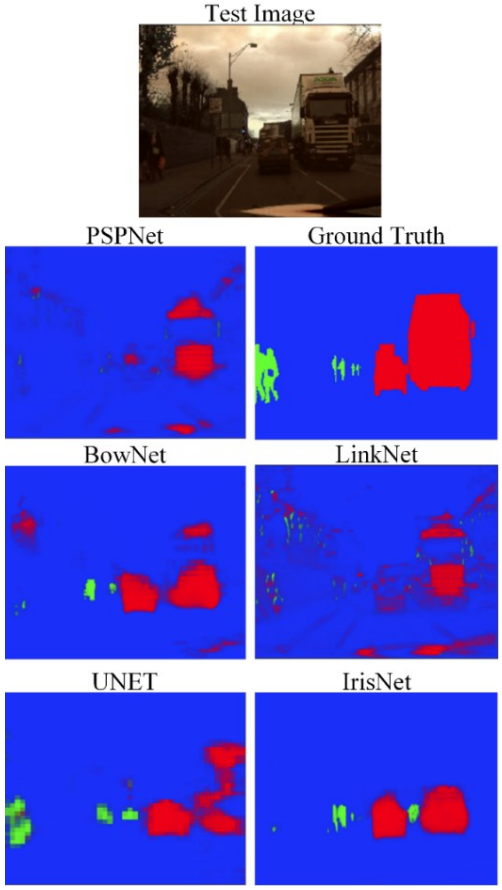



Figure 6-23. One case of the poor performance of CNN models *(Chaurasia & Culurciello, 2017; M. Hamed Mozaffari et al., 2019; Ronneberger et al., 2015; H. Zhao et al., 2017)* trained on CamVid dataset for detection of a large-scale object (truck) without using a pre-trained model.

## 6.3 A Second Language Pronunciation Training System

One of the critical aspects of the second language (L2) acquisition as an integral part of communication skills is pronunciation. From the perspective of a listener, it is often the first indication of a language learner's linguistic abilities *(Bird et al., 2018)*. Pronunciation directly affects many social interaction skills of a speaker, such as communicative competence, performance, and self-confidence. Furthermore, previous studies revealed that other aspects of L2 learning, such as word learning, can be improved by accurate pronunciation *(Johnson et al., 2018)*.

Besides the importance of L2 pronunciation, it is one of the most challenging skills to master for adult learners *(Abel et al., 2015)* in traditional classroom settings. There is often no explicit pronunciation instruction for language learners because of limited class time and lack of knowledge of effective pronunciation teaching and learning methods *(Abel et al., 2015)*. Standard practice for a language learner outside of the class is to imitate a native speaker's utterances in front of a mirror limited to lip and jaw movements along with hearing of recorded acoustic data. Without any visual feedback of a native speaker and lack of awareness of how sounds are being articulated, it is difficult for language learners to improve their skills *(Abel et al., 2015)*, especially in cases where the target sounds are not easily visible *(Bird et al., 2018)*. The positions and movements of the tongue, especially all but the most anterior part, cannot be seen in the traditional approach of listening and repeating word's pronunciations *(Heather Bliss, Burton, et al., 2017)*. Language learners can only have proprioceptive feedback of their tongue location depends on practicing sounds (vowels, liquids, or others) and the amount of contact their tongue makes with the teeth, gums, and palate *(Wilson et al., 2006)* (see Figure 6-24 for an example).

Many previous investigations in the literature of L2 pronunciation acquisition and ability have been focused on acoustic studies dealing with the sound that is produced and infer the articulation that created the sound *(Wilson et al., 2006)*. Nevertheless, both acoustic and articulatory studies are undoubtedly valuable tools for understanding the progress of an L2 learner. The latter one can often give a more accurate picture of the actions performed by the pronunciation learner while it looks directly at the articulators (e.g., the tongue, the lips, and the jaw) *(Wilson et al., 2006)*. Employing acoustic data alone might jeopardize the understanding of L2 learners in mapping the acoustic information onto articulatory movements. Having seen the articulators directly by learners, they can probably improve their pronunciation by the perception of the

articulatory adjustments *(Wilson et al., 2006)*. Therefore, an effective way of pronunciation teaching and learning includes listening and repeating using both acoustic and articulatory information.

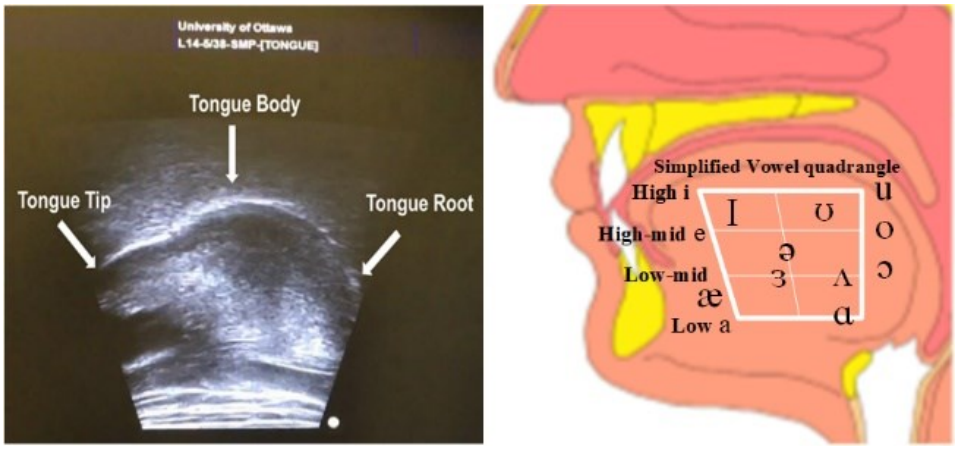



Figure 6-24. The approximate position of the tongue when producing a vowel. Sound varies depending on the position and shape of the tongue, which is not visible from outside. The tongue surface can be seen as a bright region on the ultrasound image on the left.

In this section, we explain the details of our second language pronunciation training system. Ultrasound video frames of the tongue movements are illustrated to a language learner while the tongue contour is tracked in real-time. As an augmented reality guideline and without the needs of any head fixture, the learner can move her or his head during the pronunciation training session. The correct placement of the tongue on the face view of the language learner as well as augmented the segmented tongue contour in pink color are performed fully automatically and in real-time using novel deep learning techniques. Before the explanation of our system, first, we present our preliminary investigation for proposing a pronunciation training system using ultrasound technology.

### 6.3.1 Primary System (Semi-automatic and Offline using Ultrasound)

Although ultrasound has been successfully utilized in L2 pronunciation training, interpretation of raw ultrasound data for language learners is a challenging task, especially in a real-time video stream. Moreover, to interpret ultrasound video data confidently, a language learner ought to be familiar with gestures and structures of the tongue in ultrasound data *(Bliss et al., 2016)* (see Figure 3-1 for a sample of tongue ultrasound video frame). An ultrasound-enhanced multimodal approach is a novel alternative for assisting L2 pronunciation learners in understanding and perceiving the location of their tongue on ultrasound video frames. Significant pioneer studies of this method have been accomplished by researchers in Linguistic department at the University of British

Columbia (UBC) *(Abel et al., 2015; Bird et al., 2018; Bliss, Bird, et al., 2018; Bliss et al., 2016; Gick et al., 2008; Wilson et al., 2006; Yamane et al., 2015)*. The key technological innovation of their system is manual overlaying mid-sagittal ultrasound video frames of the tongue on the external profile view of a speaker's head. This technique will allow language learners to observe a video of their speech production, projected on their face profile *(Gick et al., 2008)*. In order to highlight the tongue region in ultrasound frames, the intensity of pixels related to the tongue region was manipulated manually by a pink color.

Previous experimental results revealed the benefit of an ultrasound-enhanced multimodal approach for pronunciation language training *(Bliss, Abel, et al., 2018; Hamed Mozaffari et al., 2019; M. Hamed Mozaffari et al., 2018)*. However, manual work is extensive in many stages (pre-processing such as image enhancement, during the exam like overlaying ultrasound frames on video frames, and post-processing including highlighting of the tongue region and audio/video synchronization). Furthermore, the overlaid videos come with some non-accuracy due to the lack of transformational specification, including exact scale, orientation, and position information of the ultrasound frame for superimposing on face profile. Even exact transformations cannot be generalized for the face profile of any language learner, often restricted to one position of the head. Accurate synchronization between ultrasound data, video frames, and acoustic records is another challenge for previous studies. Besides those difficulties, a quantitative study of tongue movement only viable after freezing a target frame or during the post-processing of recorded frames *(Abel et al., 2015; B. Bernhardt et al., 2005)*.

Using ultrasound technology for pronunciation training has been exploited in different studies for many years. Recently, relatively accessible technologies such as portable ultrasound and powerful computing units enable researchers to propose novel and efficient techniques for training second language skills. Recorded videos of ultrasound tongue movements superimposed on another recorded video of a learner's face view experimented in real-class rooms with successful outcomes. In an experiment, we propose an ultrasound-enhanced multimodal L2 pronunciation training system to alleviate some weakness of previous studies and provide a better tool for other researchers in this literature. The main modules of our language pronunciation training system are illustrated in Figure 6-25. In both modules, the ultrasound data, contain tongue contour information extracted by our proposed tracking method, are overlaid automatically on a video recorded from face profile. In general, two superimposed videos are played on a computer's screen for a language learner, one from a real-time recording of him/his tongue and the

other from an instructor's tongue during speech *(Hamed Mozaffari et al., 2019; M. Hamed Mozaffari et al., 2018)*.

In previous ultrasound-enhanced methods, all steps should be accomplished manually and offline. However, in this experiment, our system to register ultrasound video on the face view of the user automatically, a face detection algorithm from the OpenCV library (Haar cascade) is employed, which can find the approximate location of the tongue video on the face. In this method, the face profile of the user is detected and using predefined approximate transformations, and the tongue video is transformed on the face of the user. Thus, students could use the system with more flexibility than previous studies without using any fixtures for fixing their head and ultrasound probe position. The sU-NET model was utilized in our system to track the tongue surface automatically and provide a guideline for the user during the training. Our system does not require any initialization steps, and due to the simplicity of our deep network architecture, GPU facility is not necessary. Figure 6-26 presents an overview of our proposed system for training second language pronunciation using ultrasound technology.

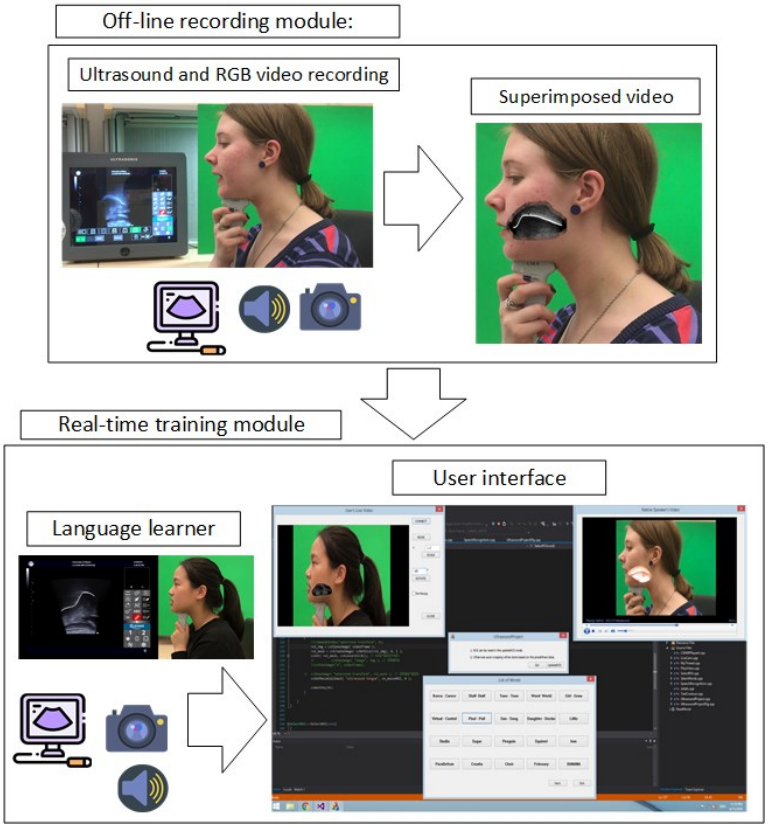



Figure 6-25. Schematic of our language training system. Language learners can see both off-line and real-time data on a computer screen. The off-line video is played with a small delay, and language learners

imitate that word. The learner comprehends the differences between the two videos and tries to duplicate an instructor's video.

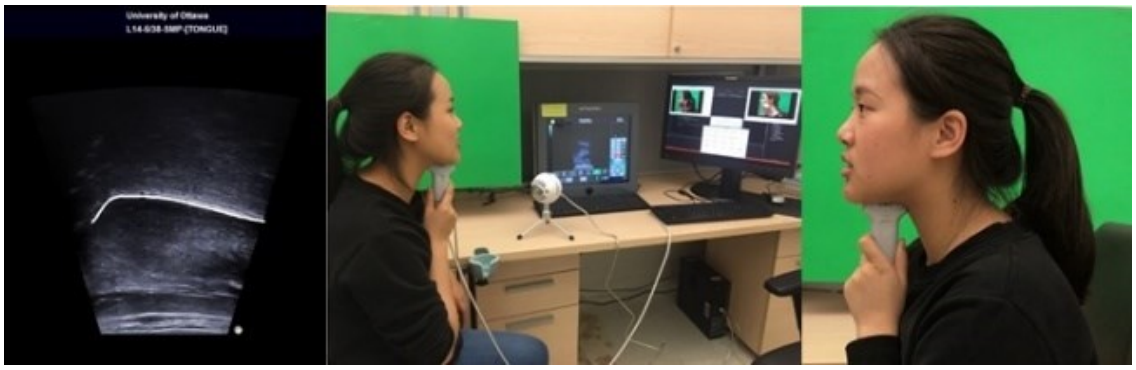


Figure 6-26. A learner is pronouncing a list of words after hearing the voice and watching the video from the instructor. The white color line is extracted from ultrasound data automatically.

### 6.3.2 Ultrasound-enhanced Multimodal Approach Freehand

Although our preliminary pronunciation training system for language learners provides more effective training, flexibility, and interactivity, still, the system is not fully automatic and accurate. The system is in real-time, but tongue video is only translated on the face, and the user is not free to move her or his head in any direction. Manual works are still necessary for cropping video data, and manipulation is vital for optimum placement of the tongue video on the face view. On the other hand, tracking the face profile is not a reliable method, and it is subjective where the performance of the system is changed from one user to another. Furthermore, there are more difficulties, such as different background scenes, face occlusions, ambient light properties, to name a few. For this reason, we implemented a new pronunciation training system to address all these issues.

Our system has two main modules for automatic and real-time tongue contour and ultrasound probe tracking using IrisNet and ProbeNet networks (see Chapter 5 for details of these deep learning models). In order to train IrisNet and ProbeNet, we used the Ultrasound dataset (chapter 4.1) and a combination of real and artificial face datasets (chapter 4.2). We used publicly available Python language and several standard libraries, including TensorFlow, as a modular system to enable other researchers for any future improvements or customization. Figure 6-27 represents a schematic of our pronunciation training modules and their connections. As can be seen in the figure, there are two streams of data recording in our system, an off-line module that is used for recording videos by

native speakers for teaching and an online module for L2 learners, which is used for pronunciation training.

Ultrasound data analysis generally involves capturing both audio and ultrasound signals together so that the audio track can help with identifying target sounds on the ultrasound data stream. Recording can be done simultaneously during recording or can be part of post-processing. Therefore, synchronization between audio and video is an integral and challenging part of having an accurate analysis. We record all data, including ultrasound video (using screen recording), camera video, and audio recording with the same frame rate (30 frames per second (f/s)). Because the performance of deep learning models is faster than the real-time recording of video data (IrisNet and ProbeNet have work faster than 30 (f/s)), there is no delay in the tracking of the ultrasound probe and tongue surface in the whole experiments.

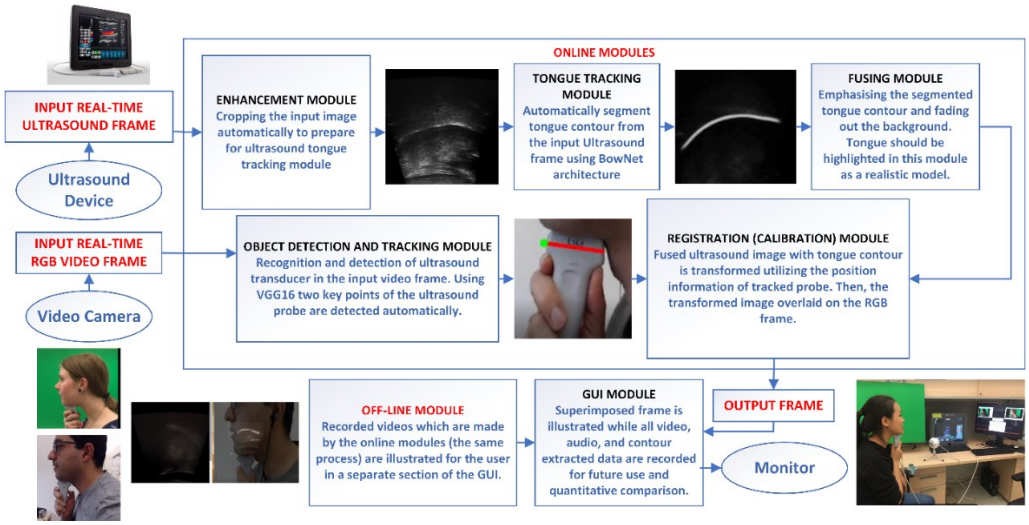



Figure 6-27. The detailed architecture of our multimodal, real-time, and automatic pronunciation training system comprises of two main online and offline modules.

Our multimodal system encompasses different data and techniques during its performance, including probe tracking, ultrasound visualization, tongue surface segmentation, tongue contour extraction, audio recording and playback, learner's lips visualization, and network connections between ultrasound and workstation. In order to have an approximately synced system, all these stages work together as the following procedure (see Figure 6-27):

1. The data stream from RGB camera (Video and Audio) is captured and visualized in real-time. We used a Logitech Webcam with a framerate of 30 fps connected to

our workstation (a personal computer with a CPU of 7 cores and 16 GB of memory equipped with a GPU GTX1080). Video and audio are already synced in this stage.

2. Ultrasound stream video data is acquired and sent to the same workstation using Microsoft Windows remote desktop software. We employed a linear transducer L14-38 connected to an Ultrasonix Tablet with settings of the tongue (depth of 7 cm, a frame rate of 30 fps). It is noteworthy to explain that instead of working on the ultrasound stream in our Python language codes, we used a Windows capturing library of OpenCV to grab ultrasound video from remote desktop software. This method enabled our system to be an ultrasound device-independent where it can work with any other ultrasound devices.
3. The current RGB Video frame is fed to our probe tracking module. The pre-trained ProbeNet model provides locations of the two defined points on the ultrasound probe. In a predefined automatic calibration process, position, orientation, and probe head length are determined automatically, and then they are sent to the visualization module. Simultaneously, the current ultrasound video frame is cropped, scaled, and fed to the deep learning model (IrisNet) for the sake of tongue region segmentation. Note that for quantitative studies, there are many methods for extracting tongue contour from the delineated tongue region, such as skeletonizing using morphological operators.
4. Results of stage 3, which are three images comprise of ultrasound frame, segmented tongue region, and RGB video frame, are superimposed using transformation information from ProbeNet. Ultrasound frame was defined as the double size of the probe head size in our ultrasound device, oriented with the angle of the probe head, and translated to centered upper-part of the probe head for some samples. Note that there are weight parameters for transparency of each image in the superimposed image. For example, the lower row of Figure 6-30 is darker than the upper ones due to more significant weight for the ultrasound image compares to the other images.
5. Superimposed video frames are sent to the visualization module. The development of this module is still in the early stages. In this module, a designed graphical user interface (GUI) in Python language would illustrate several video streams from recorded datasets, real-time monitoring, audio data analysis bar, individual frames from ultrasound and video camera, results of real-time quantitative analysis, superimposed frames, and possibly data from another ultrasound stream. There is also another camera for recording lip movements during pronunciation training. We used another webcam for this part.

In our current system, a pronunciation learner or teacher can see several windows in real-time separately on a display screen at the same time depends on the session target. For instance, as a speech investigation session, a researcher can see weighted ultrasound data superimposed on the face side in real-time, as well as separate non-overlaid ultrasound and RGB videos, accompany the segmented tongue frame. Separated data will assist researchers in comparing different tongue contours qualitatively and quantitatively with a native speaker or recorded videos from previous sessions as a follow-up study.

Figure 6-28. Screenshots from the screen of our pronunciation system from different experiments with different tongue contour color and transparency.

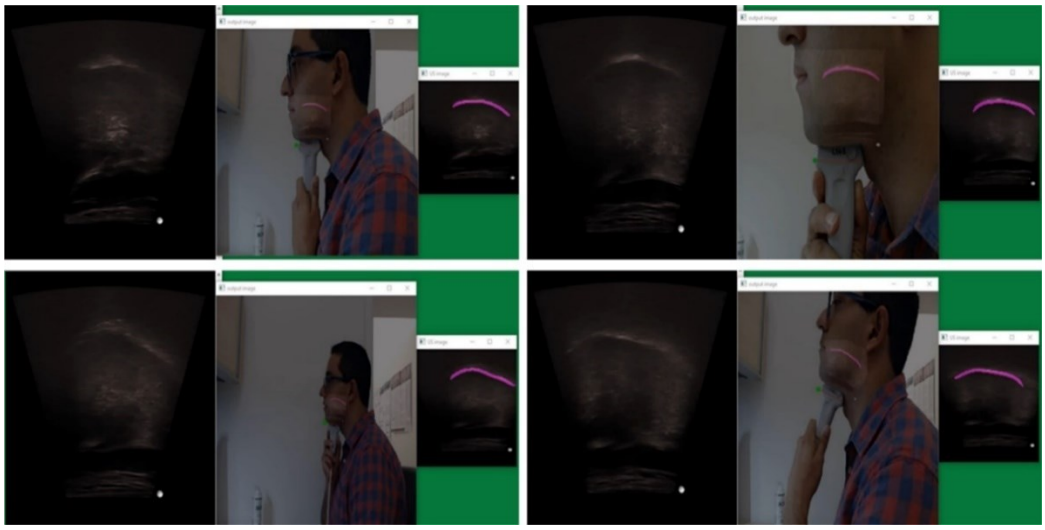


Figure 6-29. Screenshot from our pronunciation training system. IrisNet automatically tracks tongue contour in real-time. Calibration data are determined by another deep learning model to superimpose ultrasound data on the user's face.

Using our pronunciation training system and two similar ultrasound devices, L2 language teacher and learner can monitor and compare their tongues both in real-time where our system is capable of illustrating the difference between their tongue contours automatically. Due to the lack of the second ultrasound device, we used recorded videos as the second reference video for our study. It is noteworthy to mention that in previous studies, researchers should freeze one target frame for any post-processing investigation or to capture critical moments in articulations (e.g., the stop closure in a stop articulation), which requires time-consuming manual works *(Yamane et al., 2015)*.

Using triangular formulas in our calibration procedure and positional information predicted by ProbeNet, we could calculate an estimation of the correct position, orientation, and scale of ultrasound frames and segmented tongue region. In this way, ultrasound frames with augmented contour regions are transformed on the face-side of the language learner on RGB video data. Therefore, a language learner can see her or his augmented tongue movements in real-time on her or his face-side. A guideline on the probe is also shown to help the language learner to keep the probe in a correct position in two-dimensional space (see red lines and the green point in Figure 6-30). Audio data, camera, ultrasound, lip's movements, and augmented version videos, as well as tongue contour information, are simultaneously recorded for follow-up studies.

In this experiment, we proposed and implemented a fully automatic and real-time modular ultrasound-enhanced multimodal pronunciation training system using several novel innovations. Unlike previous studies, instead of tracking the user's face or using tracking devices, the ultrasound probe is tracked automatically using a deep learning model (ProbeNet), which was trained in advance on our dataset comprises of augmented real and artificial images of several L2 language learners (see chapter 4.2). This approach enables our pronunciation system to determine the optimum transformation quantities for multimodal superimposition as a user independence system. At the same time, another pre-trained deep learning model (IrisNet) extracts the tongue regions highlighted with a pink color on ultrasound data. Except for the preparation of training datasets, all modules in our system work automatically, in real-time, end-to-end, and without any human manipulation. Our preliminary experimental results revealed the significance of our system for better L2 pronunciation training.

Our pronunciation training system, fully automatic and in real-time, passed all pilot experiments. In order to show the ability of the system for use by other researchers, we used a recorded video from the University of Michigan instead of our real-time video stream from our ultrasound machine. Figure 6-28 illustrates sample frames from this

study. As can be seen, our system works even more accurate on this video because their ultrasound device was designed and set for visualization of the tongue while our ultrasound probe is not a suitable transducer for the acquisition of tongue images.

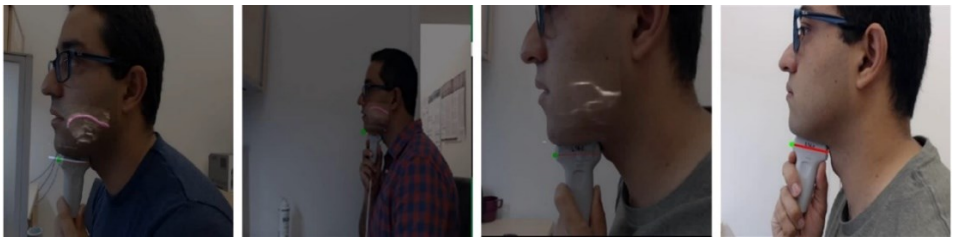


Figure 6-30. A sample frame from our real-time language training system. Note that the image is raw data for the sake of illustration, and post-processing can enhance the image significantly.

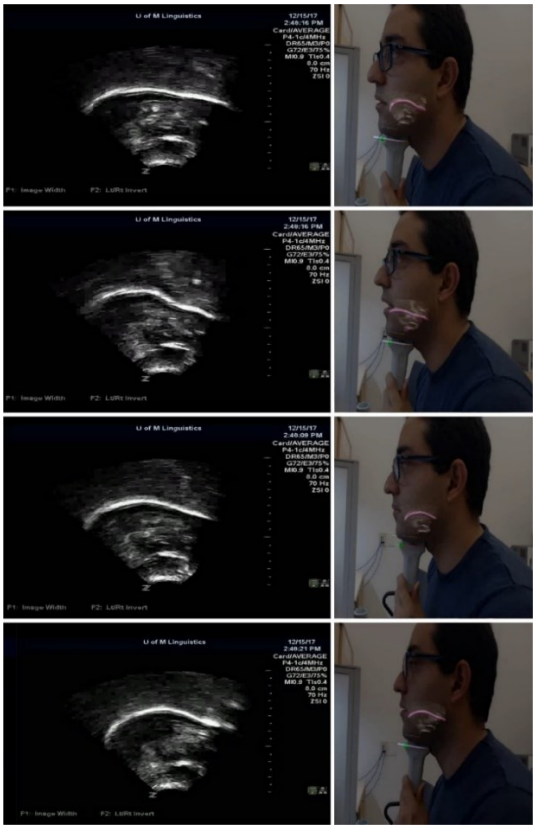


Figure 6-31. Using a sample ultrasound video from the University of Michigan to show the ability of our system in terms of generalization ability.

### 6.3.3 Ultrasound-enhanced Multimodal Approach using 3D printing

In the previous section, our pronunciation training system illustrates the tongue contour movements as augmented reality for a language learner. Although the language learner should keep the ultrasound probe in the midsagittal plan during a training session, there is no restriction to keep the probe in the middle. Therefore, there might be circumstances that the user keeps the probe in a wrong angle. For applications such as language pronunciation training, as we explained, our system will notify the user to keep the probe in the correct position. However, for linguistics studies, which researchers are looking to keep the ultrasound probe in a consistent place under the chin and the midsagittal plane during the exam session, the freehand method result is not reliable. For this reason, we designed a stabilizer that keeps the ultrasound probe under the chin during language training or tongue examination sessions.

We deployed our pronunciation training system like the freehand system using the python language and several standard public libraries as a modular system to enable other researchers for any future improvements or customization. Figure 6-27 represents a schematic of our pronunciation training modules and their connections. As can be seen in the figure, there are two streams of data recording in our system, an off-line module that is used for recording videos by native speakers for teaching and an online module for L2 learners, which is used for pronunciation training.

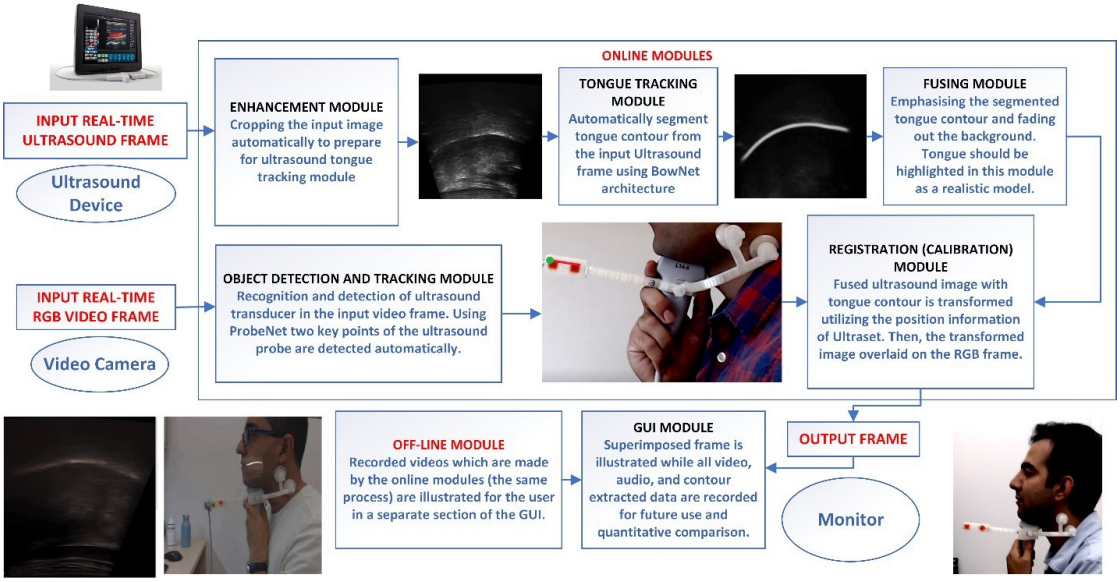



Figure 6-32. The detailed architecture of our multimodal, real-time, and automatic pronunciation training system comprises of two main online and offline modules.

Similar to our freehand system, our multimodal system encompasses different data and techniques during its performance, including probe tracking, ultrasound visualization, tongue surface segmentation, tongue contour extraction, audio recording and playback, learner's lips visualization, and network connections between ultrasound and workstation. In order to have an approximately synced system, all these stages work together as the following procedure (see Figure 6-32 for more details):

1. The data stream from RGB camera (Video and Audio) is captured and visualized in real-time. We used a Logitech Webcam with a framerate of 30 fps connected to our workstation (a personal computer with a CPU of 7 cores and 16 GB of memory equipped with a GPU GTX1080). Video and audio are already synced in this stage.
2. Ultrasound stream video data is acquired and sent to the same workstation using Microsoft Windows remote desktop software. We employed a linear transducer L14-38 connected to an Ultrasonix Tablet with settings of the tongue (depth of 7 cm, a frame rate of 30 fps). It is noteworthy to explain that instead of working on the ultrasound stream signals in our python codes, we used a window capturing library to grab ultrasound video from remote desktop software. This method enabled our system to be an ultrasound device-independent where it can work with any other ultrasound devices.
3. The current RGB Video frame is fed to our Probe tracking module. The pre-trained ProbeNet model provides locations of two key points on the 3D printable stabilizer UltraChin (for details review the Chapter 5.2.2). In a predefined automatic calibration process, position, orientation, and probe head length are determined, and then they are sent to the visualization module. Simultaneously, the current ultrasound video frame is cropped, scaled, and fed to the IrisNet model for the sake of tongue region segmentation *(M Hamed Mozaffari et al., 2019)*.
4. Results of stage 3, which are three images comprise of ultrasound frame, segmented tongue region, and RGB video frame, are superimposed using transformation information from ProbeNet and calibration calculations. Ultrasound frames are scaled, oriented, and translated using the location information of the two key points on the UltraChin markers (markers in orange color are presented in Figure 6-30 and Figure 6-31). Note that there are several weight parameters for transparency of each image in the superimposed image.
5. A superimposed video stream is sent to the visualization module. In this module, a designed graphical user interface in python language would illustrate several video streams from recorded datasets, real-time monitoring, audio data analysis bar, individual frames from ultrasound and video camera, results of real-time

quantitative analysis, and possibly data from another ultrasound stream. There is also another camera for recording lip movements during pronunciation training. We used another webcam for this part.

Head and probe stabilization is not necessary if the system is only utilized as a pronunciation bio-feedback *(Gick et al., 2005)*. However, the accuracy of our system could be improved by adding 3D printable extensions to the UltraChin for head stabilization for a particular linguistic study, while articulatory data will be subject to detailed quantification and measurement. In order to evaluate our 3D printable design, we followed the method in *(Scobbie et al., 2008)*. We attached one magnetic tracking sensors, PATRIOT Polhemus Company (see Figure 6-33), on the UltraChin and participant's chin in two separate experiments. Six degrees of freedom (see Figure 6-34) was recorded after ten times repeating a similar experiment. For this experiment, the participant's head was fixed using the method in *(Ménard & Noiray, 2011)*. We asked the participant to repeat "ho-mo-Maggie" *(Scobbie et al., 2008)* and to open mouth to the maximum position for ten times. We calculated deviations of the UltraChin in terms of transnational (in millimeters) and rotational (in degree) slippages.

Table 6-17 shows the maximum error of the UltraChin after ten times of experiment. For a better understanding of the UltraChin performance, we checked two different settings where four screws of the device were loose (most comfort) or tight (discomfort).

Table 6-17. Maximum slippage of the UltraChin in 6 DOF after ten times testing on one participant. Values show the mean and standard deviation for each experiment. Screws of the UltraChin was loosely and tightly firm in two different experiments.

| Status of four screws | Max translational in millimeters | | | Max Rotational in degree | | |
|---|---|---|---|---|---|---|
| | x | y | z | roll | yaw | pitch |
| Loose | 4.7±0.39mm | 5.1±0.69mm | 7.6±0.69mm | 6.4±0.21° | 4.1±0.46° | 5.9±0.86° |
| Tight | 3.4±0.18mm | 3.5±0.72mm | 6.1±0.15mm | 5.6±0.59° | 3.8±0.45° | 4.7±0.91° |

Our experimental results showed that in the case of tightly firming four screws of UltraChin user's chin has a better long term translational and rotational unwanted slippage without losing a significant comfortability for the user's neck. Slippage errors might be even more due to the usage of cushions, skin deformations, and how the participant is keeping the probe under the chin. In comparison to the system in *(Scobbie et al., 2008)*, UltraChin has more long-term slippage in almost all directions. One reason

is that UltraChin has fewer stabilizers than previous helmets. Nevertheless, UltraChin errors still are within acceptable deviation limits reported in *(Scobbie et al., 2008)*.


Figure 6-33. Illustration UltraChin connected to Electromagnetic Sensor. The receiver of the Electromagnetic tracker is attached to the camera. ProbeNet tracks the upper left corners of the two orange squared markers for the calculation of transformational information. The electromagnetic tracker is Polhemus Patriot TM. First row: different views of magnetic tracking sensors attached to UltraChin. Second row: Different parts of UltraChin and magnetic tracking sensor can be seen in the figure.

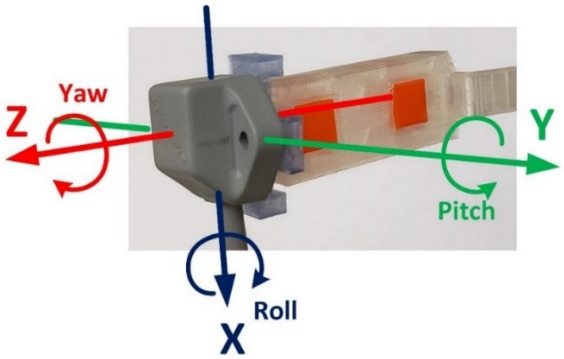



Figure 6-34. The electromagnetic transmitter sensor is screwed to the UltraChin for evaluation accuracy of our system. The sensor provides six degrees of freedom in real-time, including Yaw, Pitch, and Roll.

Similar to our freehand system, using UltraChin, a language pronunciation learner or teacher can see several windows in real-time separately on a display screen at the same time depends on the session target. For instance, as a speech investigation session, a researcher can see weighted ultrasound data superimposed on the face side in real-time, as well as separate non-overlaid ultrasound and RGB videos, accompany segmented ultrasound frames. Separated data will assist the researcher in comparing different tongue contours qualitatively and quantitatively with a native speaker or recorded videos from previous sessions as a follow-up study.

Due to independence characteristic of our system respect to the number of image processing streams as a multimodal system, in a different scenario, using two similar ultrasound device, L2 language teacher and learner can see and compare their tongues in real-time, and our system is capable of illustrating the difference between their tongue contours automatically. Due to the lack of the second ultrasound, we used recorded videos as the second reference video for our comparison study. It is noteworthy that in previous studies, researchers should freeze one target frame for any post-processing investigation or to capture critical moments in the articulation (e.g., the stop closure in a stop articulation), which requires time-consuming manual works *(Yamane et al., 2015)*.

We proposed several modules employing two different deep learning models for automation purposes in this experiment. For the ultrasound contour tracking module, we used the IrisNet model similar to our Freehand system (see Chapter 6.1.4 and 6.3.3 for more details about the training and testing of IrisNet model). In this section, we focus more on the usage of UltraChin for the linguistic perspective applications. In order to train ProbeNet for tracking the markers on the UltraChin, we created a dataset comprises of 600 images of 3 different participants. Recorded frames are annotated manually by placing two pre-defined key points on the upper-left side of orange markers. Dataset was divided into 80% training, 10% validation, and 10% testing sets.

Using our data augmentation toolbox (applying rotation, scaling, translation, and channel shift for images and the two key points), we created a dataset of 5000 images, and their corresponding annotated data. Adam optimization algorithm was utilized for training of the ProbeNet with the first and second momentum of 0.9 and 0.999, respectively. For training and validation purposes, the loss function is Mean Absolute Error (MAE), where the learning rate is variable with an exponential decay rate of $10^{-5}$ and an initial value of 0.001. We trained the ProbeNet model for ten epochs with mini-batches of 10 images.

During the video recording session, employing the UltraChin device, the ultrasound probe is kept by the user under the chin. UltraChin guarantees that the ultrasound probe is in the Midsagittal plane with acceptable accuracy. Furthermore, the ProbeNet model tracks the markers on the UltraChin, results in user independence characteristic for the system. Note that UltraChin is designed as a general device attachable for all types of ultrasound probes. Our experimental results revealed an MAE of 0.027±0.0063 for the accuracy of ProbeNet in the tracking of UltraChin markers.

In general, a collaboration between UltraChin device and ProbeNet as well as predefined calibration calculations, our language pronunciation training system is capable of illustrating tongue movements in real-time and automatically in augmented reality on users' faces. In comparison with our freehand system, using the UltraChin device increases the accuracy of tracking and quantitative tongue contour studies. However, the flexibility of users in terms of neck movement drops considerably. Figure 6-35 and Figure 6-36 present a few sample frames from our language pronunciation training system.


Figure 6-35. A sample frame from our real-time language training system. Note that the image is raw data for the sake of illustration, and post-processing can enhance the image significantly.

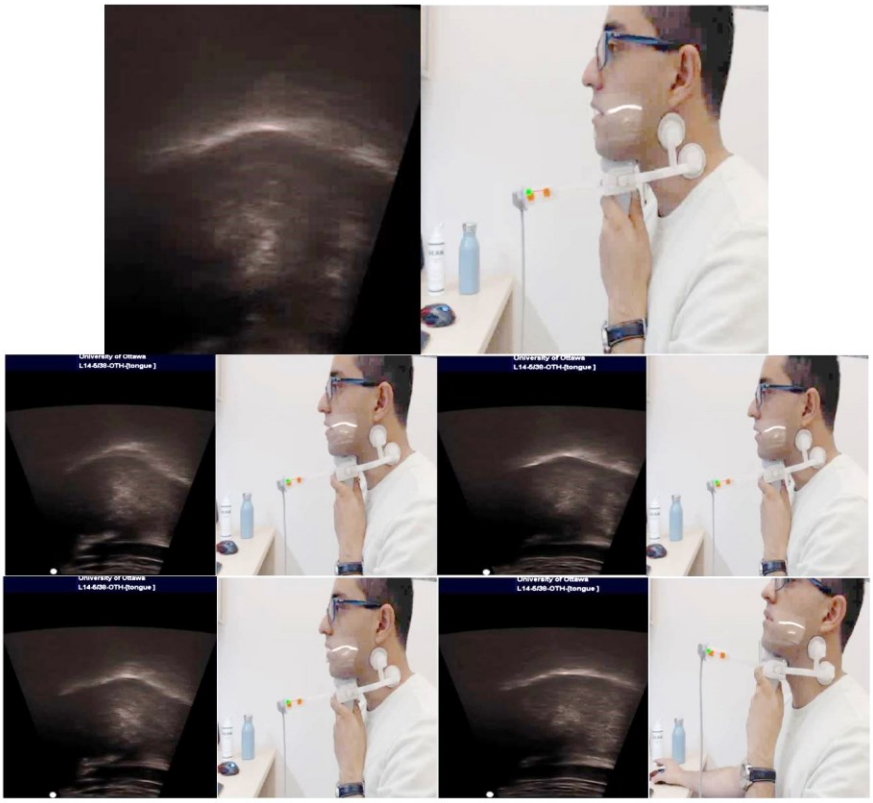


Figure 6-36. Tracking of the ultrasound probe in real-time. Two ends of the red lines show the place of two key points on the probe. Green point is considered for the sake of ultrasound frame calibration.

### 6.3.4 User Study of the Second Language Pronunciation Training

**Preliminary Study:** In order to test the potential utility of our language training system in the second language, a preliminary investigation was conducted with two Chinese students as participants. Three English native speakers were considered as instructors and trained to read predefined words. Recording procedures is a two-fold process (see Chapter 6.3.1 for more details). Off-line recording module (see Figure 6-37) in which native speakers read a list of predefined difficult words. In this experiment, a list of two kinds of difficult words: pairs and single words (see Table 6-18) was produced by conducting a survey using questionnaires among 20 Chinese students at the University of Ottawa, asking about the words which they have more difficulties in pronouncing in English.

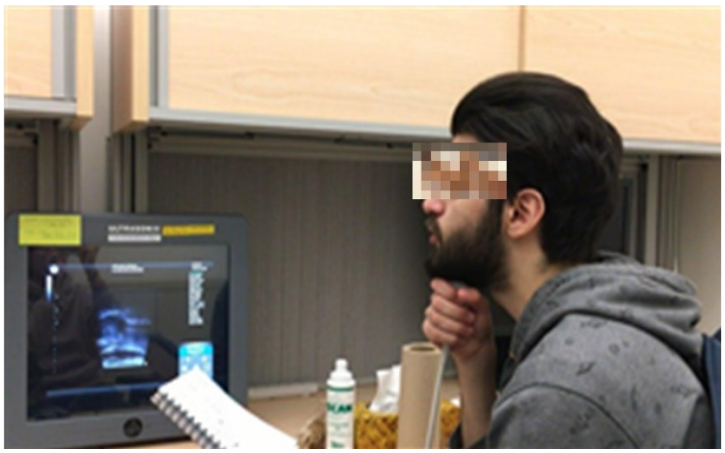


Figure 6-37. An instructor is pronouncing a list of words, and his face and voice are recorded by the camera as well as ultrasound video. Manually, cropped ultrasound video and the tongue contour extracted (green line) from that video are superimposed with the video from the camera.

Table 6-18. A set of words with difficulty for Chinese second language learning.

| Pairs | Korea<br>Career | Stuff<br>Staff | Tone<br>Tune | Word<br>World | Girl<br>Grow | Virtual<br>control | Pool<br>pull |
|---|---|---|---|---|---|---|---|
| Single word | Little | Studio | Sugar | | | | |

Our preliminary experimental results *(Hamed Mozaffari et al., 2019; M. Hamed Mozaffari et al., 2018)* indicate the usability of our semi-automatic system in comparison with previous offline systems. Experience of language learners was that perceiving their tongue movements by superimposing video on their face is more efficient in pronunciation learning than watching recorded videos from a different person even in multimodal versions such as *(Abel et al., 2015)*.

In order to prove this conclusion, we conducted two different user studies experiments using our freehand pronunciation training system.

**User study 1:** There are many methods and standards in the literature for testing L2 pronunciation acquisition methods *(Adler-Bock et al., 2007; B. Bernhardt et al., 2005; Gick et al., 2008)*. It is possible to use the system for small numbers of L2 pronunciation training individuals *(Gick et al., 2008)* in a large classroom setting, either by providing individual ultrasound training to language instructors *(Noguchi et al., 2015)* or by presenting ultrasound videos as part of a blended learning approach *(Bliss, Cheng, et al., 2017)*, or even in a community-based settings *(Bird et al., 2018)*. For example, in *(Yamane et al., 2015)*, the previous ultrasound-enhanced system has been tested in several courses at UBC. This kind of system is also tested for training different indigenous languages for

learning and revitalization *(Bliss et al., 2016)*. The usability of ultrasound bio-feedback in L2 pronunciation training has been comprehensively investigated in *(Antolik et al., 2019; B. Bernhardt et al., 2005)*.

Figure 6-38. A learner is pronouncing a list of words after hearing the voice and watching the video from the instructor. The white color line is extracted from ultrasound data automatically.

Due to the lack of assessment facilities like a big classroom with ultrasound machines, we followed the method in *(Gick et al., 2008)* for single participant design. In this method, an approach is used repeatedly to measure the dependent variables from an individual. The dependent variables in this methodology consist of targets to be learned, such as vowels, consonants, or suprasegmental *(Gick et al., 2008)*. Therefore, the main goals of this kind of experiment include articulator position and accuracy of segments, and speech intelligibility and accuracy of production. Good candidates for ultrasound biofeedback are usually vowels, rhotic sounds, retroflex, velar and uvular consonants, and dynamic movements between tongue gestures *(Bird et al., 2018)*.

In subsequent studies, ultrasound was used to teach individual challenging sounds, such as English /r/in clinical settings *(Adler-Bock et al., 2007; Tateishi & Winters, 2013)*. For this reason, we selected one individual participant to practice predefined letters individually by comparing them with the pronunciation of the same statements by a

native speaker. We utilized sample testing videos from the enunciating website UBC language department *("enunciate UBC," 2019)* as our truth pronunciation references. An Iranian L2 pronunciation learner volunteered to use our multimodal pronunciation system for ten sessions to improve pronunciation of /r/ sound.

Each session contained 20 times repeating of /ri/, /ra/, and /ru/ and comparing with the video downloaded from *("enunciate UBC," 2019)*. Before the first session, we trained the participant for using our system and watching training videos from the same website. In order to compare sessions, we selected five points manually on tongue contours of both the subject and the native videos for a specific frame of /r/ video data. The average of absolute distances between the five points provides us a criterion (see Figure 6-39). We asked the subject to focus on video and try to mimic the tongue contours similar to the native video.

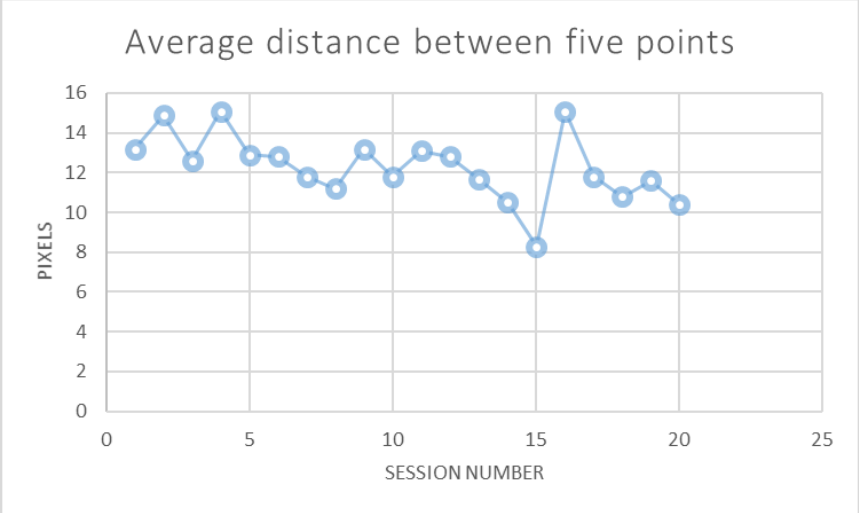



Figure 6-39. Average distances between five points on a specific frame of native and subject videos when they repeat /r/ in the English language.

The feedback from the language learner in the final session was outstanding. We investigated the participant's awareness of controlling tongue body gestures. Results show that we gained in performance after a short training session, and this suggests that providing visual feedback, even a short one, improves tongue gesture awareness. From Figure 6-39, we can see an average improvement in the pronunciation of letter /r/ for the subject, but it cannot be generalized for all subjects. An extensive pedagogical investigation of our pronunciation training system for teaching and learning should be

accomplished to evaluate the effectiveness of our system in different aspects of pronunciation training.

The benefits of our pronunciation system are not limited to only real-time and automatic characteristics. During pronunciation sessions, unlike other studies *(Hueber, 2013)*, there is no need for any manual synchronization. Furthermore, ultrasound frames, RGB video frames, audio, overlaid images, and extracted contour information are recorded and visualized simultaneously. Our preliminary assessments showed that a language learner would fatigue slower than previous studies, in which the average time was 20 to 30 mins due to maintaining a relatively constant position *(Gick et al., 2008)*. In our system, non-physical restrictions such as using uniform backgrounds in video recording have been addressed, and the system can be used in any room with different ambient properties.

**User Study 2:** In a separate experiment, we invited 18 volunteered from Intelligence, Innovation, and Image processing Lab (IIIL) members. We first present a training video, including the introduction and history of the project, as well as the benefits of using new technologies in second language pronunciation training. Following the training session, subjects observe a demo of using the system by the principal investigator. The ethics approval of this user study can be seen in the Appendix section. After the demo session, each subject uses our freehand pronunciation system for learning the word "car". Subjects watch a native speaker's video simultaneously repeating word car using an ultrasound device. Each session took around 10 mins, where several videos and audio data were recorded from each subject, including screen recording, video from all cameras. We interviewed each subject during the training session, and we asked them to speak in their mother language for two minutes by introducing themselves and explaining their experience of using our pronunciation training system.

Following the training session, each subject answered several designed questions in a survey file where questions are is more related to their opinion about the system and their experience. The results of the survey questions are tabulated in Table 6-19. As can be seen from the table, all scores are more than 7 for our questions. In this experiment, the average of participants (44% and 56% females and males) age was 28.5 years with the youngest and oldest subjects of 22 and 49, respectively. The mother language of subjects is Farsi 22%, Mandarin 67%, Somali 5.5%, and Bangla 5.5%. 89% of subjects participated at least on time in one standard English exam such as IELTS or TOEFL. Converting their scores to the IELTS exam, their average score was 6.77 out of 9 with the minimum

and maximum of 6.5 and 7.5, respectively. Sample images of subjects during pronunciation training using our freehand system are presented in Figure 6-40.

Table 6-19. Average score from the answer of subjects to our survey questions.

| No. | Questions | Score |
|---|---|---|
| 1 | Can our proposed system improve your English pronunciation skills in the future? (10 = strongly agree, 1 = strongly disagree) | 8.08 |
| 2 | How much improvement have you experienced in learning the target letter in today's experience? (10 = strongly improved, 1 = no any improvement) | 7.15 |
| 3 | If our system is developed employed for training difficult L2 pronunciation of a difficult word in the classroom setup, our system can improve the effectiveness of student's learning and training trends? (10 = strongly agree, 1 = strongly disagree) | 8.00 |
| 4 | Watching training videos and the demo of system usage by the investigator before the training session impact positively my understanding of the tongue movements? (10 = strongly agree, 1 = strongly disagree) | 9.38 |
| 5 | Using our system is convenient for you to work with in terms of keeping the probe under your chin and watching your tongue movements? (10 = strongly agree, 1 = strongly disagree) | 7.54 |
| 6 | The ultrasound gel is convenient to use, and I can feel the gel for a long time without any discomfort? (10 = strongly agree, 1 = strongly disagree) | 8.54 |
| 7 | The accuracy of the ultrasound probe and tongue contour tracking is acceptable for focusing on learning the target word pronunciation instead of watching raw ultrasound data? (10 = strongly agree, 1 = strongly disagree) | 7.92 |
| 8 | From 1 to 10, if you want to use this system for a long period, how many minutes can you keep the probe under your chin? 1 = 5 mins, 5 = 25 mins, 10 is 50 mins | 7.54 |
| 9 | If the cost of our language training system is $500, you purchase one for self-training? (10 = strongly agree, 1 = strongly disagree) | 7.54 |
| 10 | If there is an L2 course named ("Improve your pronunciation skill by a new method ") that is facilitated by our system for each student for $500 each semester, you participate in that course? (10 = strongly agree, 1 = strongly disagree) | 8.69 |
| 11 | The impact of a real-time pronunciation training system on the training progress of L2 learners in comparison to just watching recorded videos of native speaker's speech is significant? (10 = strongly agree, 1 = strongly disagree) | 9 |
| 12 | You experienced situations that you hear the pronunciation of a new word and try to repeat, but you are not sure about your accuracy. In the previous studies, you could watch a recorded video of a native speaker. Our real-time system can help you to make sure about your pronunciation accuracy and confidence? (10 = strongly agree, 1 = strongly disagree) | 8.31 |
| 13 | In general, it is difficult for you to learn the pronunciation of a new word in the English language? (10 = strongly agree, 1 = strongly disagree) | 7.54 |
| 14 | Watching my ultrasound tongue movements augmented on my face is more effective for pronunciation training than watching the same videos separately. (10 = strongly agree, 1 = strongly disagree) | 9.31 |

We also asked the participants, general questions about their opinions about the training session and any suggestions for improving the system (see Table 6-20).

Table 6-20. General questions about the system.

| Questions | Average Results |
|---|---|
| Have you learned about your tongue gestures using ultrasound today's session? | 100% |
| Is the system works for second language pronunciation training? | 100% |
| Do you feel and perceive your tongue position in your mouth during this session? | 100% |
| Have you ever had any problem with the pronunciation of a problematic word in the past? | 77% |
| Do you think that our system can be useful for training the pronunciation of your mother language? | 54% |

In conclusion, from our user study results, we understood that our pronunciation training system could be utilized for improving the training trend of language learners by providing real-time visual feedback using the augmented reality of ultrasound data from tongue movements. The general opinion of subjects about our system was positive, where many of the subjects claimed to watch their tongue movements were for the first them. Subjects compared our ultrasound videos with data from other institutes, and they suggest using a better ultrasound probe designed for tongue visualization for improving the effectiveness of our language pronunciation system.

Application of our system can even be studied as a visual biofeedback (VBF) for other applications like pronunciation learning in different languages or development speech disorders (SSDs), which is a common communication impairment in childhood who consistently exhibit difficulties in the production of specific speech sounds in their native language *(Ribeiro et al., 2019)*. Testing the efficacy of our real-time automatic ultrasound-enhanced multimodal pronunciation system remains in the early stages. For this reason, we conducted a comprehensive user study to see the future of our freehand pronunciation language training system. Further research should be accomplished to create a fuller and more accurate assessment of our system with the collaboration of linguistic departments.

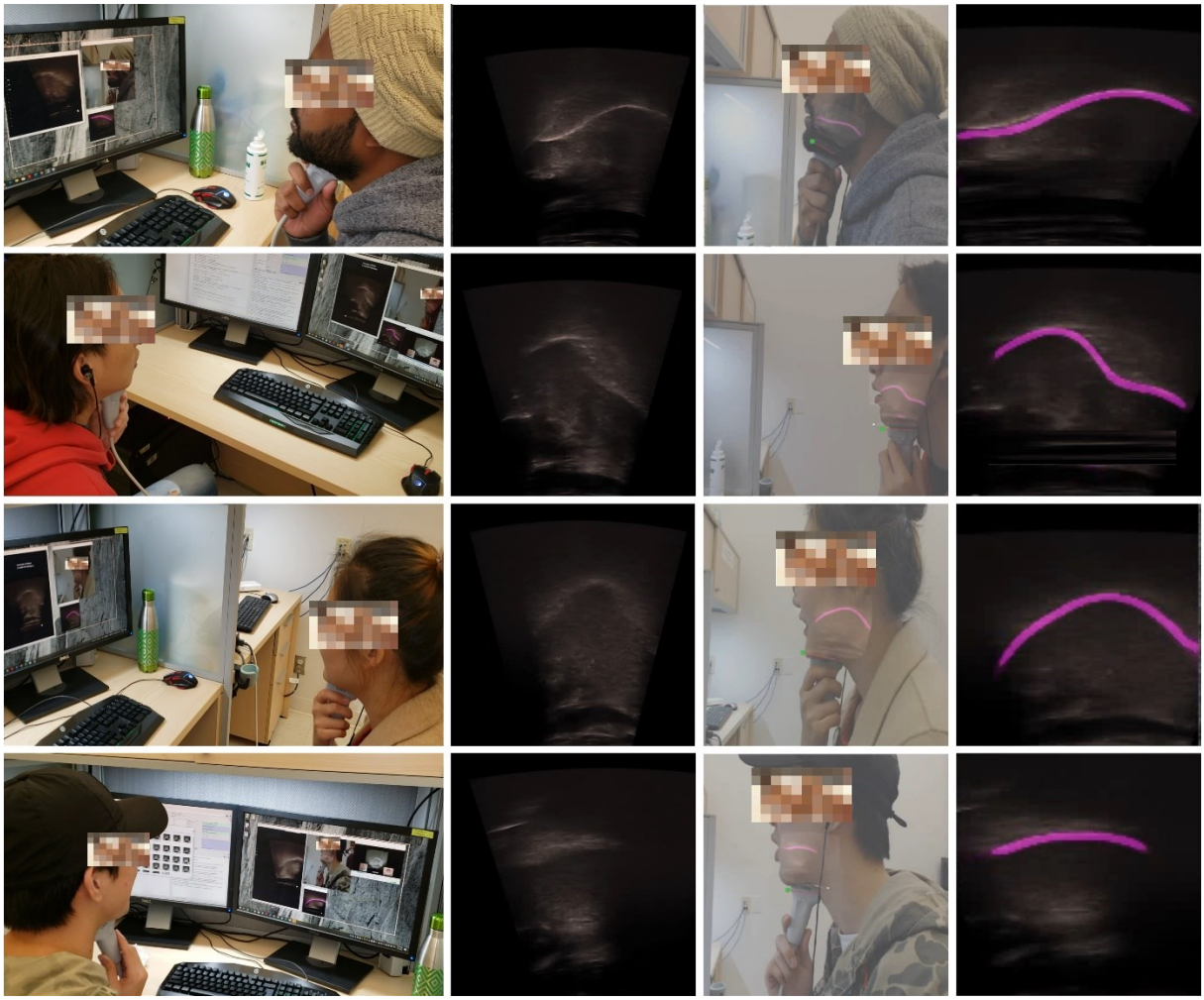


Figure 6-40. Sample images from our user study pronunciation training session using our freehand system.

# Chapter 7 Discussion and Conclusion

While the potential benefits of articulatory feedback using ultrasound data for language pronunciation have long been acknowledged, until recently, with the advancement of deep learning techniques, real-time applications are feasible for easier guided language training. In this thesis, we proposed and implemented a fully automatic and real-time modular ultrasound-enhanced multimodal pronunciation training system using several novel innovations. Unlike previous studies, instead of tracking the user's face or using tracking devices, the ultrasound probe position and orientation are estimated automatically using ProbeNet, a designed deep learning model, and predefined calibration transformational settings. UltraChin, a 3D printable stabilizer, is used as a backup for evaluation assessment of our system. To assist pronunciation training instead of manual highlighting the tongue surface, tongue contours are automatically tracked in ultrasound video data using our advanced deep learning models, including sU-NET, BowNet, and IrisNet, each of which with a new ability on ultrasound tongue data. In general, two key points on the ultrasound probe are tracked, and segmented ultrasound frames are automatically augmented on the face view of a language learner in real-time.

In contrast with other research fields with large datasets, the size of a typical ultrasound tongue dataset is no more than *200K* frames, and it makes more sense to fine-tune one model on several datasets to find the knowledge balance point as a general model for use in real-time applications on various ultrasound devices. However, due to the similar distribution of tongue datasets, it is still feasible to design a general deep learning model applicable to all ultrasound tongue datasets. The generalization ability of our proposed models was tested on several datasets from different ultrasound machines using the domain adaptation approach. In transfer learning literature, researchers usually focus on finding a decent performance on a source domain. Then they try domain adaptation from source to target task. However, a reliable and general approach is the one that can provide satisfactory results in the opposite path from target to source domain as well without compromising the reverse path.

In terms of automatic tongue contour tracking, after extensive evaluation research, we proposed and developed different deep learning models, including sU-NET, BowNet, and IrisNet, which is our latest architecture. IrisNet model with few trainable parameters could predict better tongue contour instances. Performance of IrisNet improved because of using our new convolutional module RetinaConv employing dilated and standard

convolutions simultaneously. RetinaConv, inspired by the peripheral vision ability of human eyes, emphasizes general and details the context of input feature maps using two different receptive field sizes. The primary motivation behind IrisNet architecture using the RetinaConv module was the need to implement an efficient deep learning model for semantic segmentation, which works independently from pre-trained models while capable of applying on several types of datasets. Following semantic segmentation literature, we considered background information as a separate class label in the training and validation process of our models. The consequence was the improvement of all deep learning models in the ability of discrimination between background artifacts and the region of interest, which is tongue contour. Our experimental results illustrate the powerfulness of the IrisNet model in terms of real-time performance. IrisNet can even predict excellent instances for a novel ultrasound tongue dataset without any fine-tuning.

The benefits of our pronunciation system are not limited to only real-time and automatic characteristics. During pronunciation sessions, unlike previous studies, there is no need for any manual synchronization. Furthermore, ultrasound, camera, audio, augmented, and extracted data are all recorded and visualized simultaneously. Our preliminary assessments showed that a language learner would fatigue slower than previous studies, in which the average time was 20 to 30 mins due to maintaining a relatively fixed position. In our system, non-physical restrictions such as using uniform backgrounds in video recording have been addressed, and the system can be used in any room with different ambient properties.

Application of our system can even be studied as a visual biofeedback (VBF) for other applications like pronunciation learning in different languages or development speech disorders (SSDs) which is a common communication impairment in childhood who consistently exhibit difficulties in the production of specific speech sounds in their native language (Ribeiro et al., 2019). UltraChin makes the system universal for any type of ultrasound probe as well as invariant respect to the probe image occlusion. UltraChin errors due to the slippage of the device during a speech was within the standard range in the literature. Our preliminary experimental results and conducting a user study on our second language training system revealed the significance of our system for better pronunciation learning trends. An extensive pedagogical investigation of our pronunciation training system for teaching and learning should be accomplished to evaluate the effectiveness of our system in different aspects of pronunciation training. We believe that publishing our datasets, annotation package, and our proposed deep learning architectures, all implemented in multiplatform python language with an easy to use

documentation can help other researchers in this field to fill the gap of using previous methods where several non-accessible requirements are needed as well as they customized for restricted datasets. Our novel idea of point tracking using deep learning for ultrasound tongue video data can be an alternative for all current conventional methods where available annotated datasets can be combined as a substantial standard dataset for training deep learning models. Our qualitative results were promising for this idea, but extensive assessment quantitative study should be conducted to improve the current results. Evaluation of IrisNet for other medical ultrasound image segmentation tasks can be another future of this work.

# Chapter 8 References


Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., … Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. *Methods in Enzymology*, *101*(C), 582–598. https://doi.org/10.1016/0076-6879(83)01039-3

Abel, J., Allen, B., Burton, S., Kazama, M., Kim, B., Noguchi, M., … Gick, B. (2015). Ultrasound-enhanced multimodal approaches to pronunciation teaching and learning. *Canadian Acoustics*, *43*(3).

Adler-Bock, M., Bernhardt, B. M., Gick, B., & Bacsfalvi, P. (2007). The use of ultrasound in remediation of North American English/r/in 2 adolescents. *American Journal of Speech-Language Pathology*, *16*(2), 128–139. https://doi.org/10.1044/1058-0360(2007/017)

Akgul, Y. S., Kambhamettu, C., & Stone, M. (1998). Extraction and tracking of the tongue surface from ultrasound image sequences. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 298–303. https://doi.org/10.1109/CVPR.1998.698623

Akgul, Y. S., Kambhamettu, C., & Stone, M. (1999). Automatic extraction and tracking of the tongue contours. *IEEE Transactions on Medical Imaging*, *18*(10), 1035–1045. https://doi.org/10.1109/42.811315

Alison, N., & Boukerroui, D. (2006). Ultrasound image segmentation: a survey. *IEEE Transactions on Medical ImagingIEEE Transactions on Medical Imaging*, *25*(8), 987–1010.

Allan, P., Weston, Michael, Baxter, G. M. (2011). *Clinical Ultrasound* (3rd edition). Elsevier.

Altaf, F., Islam, S. M. S., Akhtar, N., & Janjua, N. K. (2019). Going Deep in Medical Image Analysis: Concepts, Methods, Challenges, and Future Directions. *IEEE Access*, *7*, 99540–99572. https://doi.org/10.1109/access.2019.2929365

Aminur, M., Ratul, R., Mozaffari, M. H., Parimbelli, E., Lee, W., Ratul, M. A. R., … Lee, W. (2019). Atrous Convolution with transfer learning for Skin Lesions Classification. *BioRxiv Cancer Biology*. https://doi.org/10.1101/746388

Antolik, T. K., Pillot-Loiseau, C., Kamiyama, T., Antolík, T. K., Pillot-Loiseau, C., & Kamiyama, T. (2019). The effectiveness of real-time ultrasound visual feedback on tongue movements in L2 pronunciation training. *Journal of Second Language Pronunciation*, *5*(1), 72–97. https://doi.org/10.1075/jslp.16022.ant

Aslan, E., & Akgul, Y. S. (2019). *Tongue Contour Tracking in Ultrasound Images with Spatiotemporal LSTM Networks* (Vol. 2). https://doi.org/10.1007/978-3-030-33676-9_36

Avidan, S. (2004). Support Vector Tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(8), 1064–1072. https://doi.org/10.1109/TPAMI.2004.53

Ayoub, I., & Al Osman, H. (2019). *Memory-Efficient Backpropagation for Recurrent Neural Networks*. https://doi.org/10.1007/978-3-030-18305-9_22

B, C. L., Reiter, A., & Hager, G. D. (2016). *European Conference on Computer Vision*. *9905*, 36–52. https://doi.org/10.1007/978-3-319-46448-0

Babenko, B., Yang, M. H., & Belongie, S. (2011). Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(8), 1619–1632. https://doi.org/10.1109/TPAMI.2010.226

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(12), 1–14. https://doi.org/10.1109/TPAMI.2016.2644615

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). Greedy Layer-Wise Training of Deep Networks. *Advances in Neural Information Processing Systems*, 153–160.

Bergstra, J., Bengio, Y., Bergstra, James, & Bengio, Y. (2012). Random search for hyper-parameter optimization. In *Journal of Machine Learning Research* (Vol. 13). https://doi.org/10.1162/153244303322533223

Bernhardt, B., Gick, B., Bacsfalvi, P., Adler-Bock, M., & Adler-Bock, M. (2005). Ultrasound in speech therapy with adolescents and adults. *Clinical Linguistics and Phonetics*, *19*(6–7), 605–617. https://doi.org/10.1080/02699200500114028

Bernhardt, M. B., Bacsfalvi, P., Adler-Bock, M., Shimizu, R., Cheney, A., Giesbrecht, N., … Radanov, B. (2008). Ultrasound as visual feedback in speech habilitation: Exploring consultative use in rural British Columbia, Canada. *Clinical Linguistics & Phonetics*, *22*(2), 149–162.

Berry, J., & Fasel, I. (2011). Dynamics of tongue gestures extracted automatically from ultrasound. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 557–560. https://doi.org/10.1109/ICASSP.2011.5946464

Berry, J. J., & James, J. (2010). *Machine Learning Methods for Articulatory Data*.

Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., & Torr, P. H. S. (2016). Fully-convolutional siamese networks for object tracking. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9914 LNCS*, 850–865. https://doi.org/10.1007/978-3-319-48881-3_56

Bird, S., Gick, B., on, B. G.-P. Isap. 2018 I. S., 2018, undefined, & Gick, B. (2018).

Ultrasound biofeedback in pronunciation teaching and learning. *ISAPh 2018 International Symposium on Applied Phonetics*, (September), 5–11. https://doi.org/10.21437/ISAPh.2018-2

Bliss, H., Abel, J., & Gick, B. (2018). Computer-assisted visual articulation feedback in L2 pronunciation instruction. *Journal of Second Language Pronunciation*, *4*(1), 129–153. https://doi.org/10.1075/jslp.00006.bli

Bliss, H., Bird, S., Ashley Cooper, P., Burton, S., & Gick, B. (2018). Seeing Speech: Ultrasound-based Multimedia Resources for Pronunciation Learning in Indigenous Languages Licensed under Creative Commons Attribution-NonCommercial 4.0 International Seeing Speech: Ultrasound-based Multimedia Resources for Pronunciation Lear. *University of Hawaii Press*, *12*, 315–338. Retrieved from http://nflrc.hawaii.edu/ldchttp://hdl.handle.net/10125/

Bliss, H., Burton, S., Gick; Bryan, & Bryan, G. (2017). Using Multimedia Resources to Integrate Ultrasound Visualization for Pronunciation Instruction into Postsecondary Language Classes. *Journal of Linguistics and Language Teaching*, *8*(2), 173–188. Retrieved from https://sites.google.com/site/linguisticsandlanguageteaching/home-1/volume-8-2017-issue-2/volume-8-2017-issue-2---article-bliss-et-al

Bliss, H., Burton, S., & Gick, B. (2016). Ultrasound overlay videos and their application in Indigenous language learning and revitalization. *Canadian Acoustics*, *44*(3).

Bliss, H., Cheng, L., Schellenberg, M., Lam, Z., Pai, R., & Gick, B. (2017). Ultrasound Technology and its Role in Cantonese Pronunciation Teaching and Learning Background : *Proceedings of the 8th Annual Conference on Pronunciation in Second Language Learning & Teaching.*

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, R. (1994). Signature Verification using a "Siamese" Time Delay Neural Network. *In Advances in Neural Information Processing Systems*, 737–744.

Brostow, G. J., Shotton, J., Fauqueur, J., & Cipolla, R. (2008). Segmentation and Recognition Using Structure from Motion Point Clouds. *ECCV (1)*, 44–57.

Caelles, S., Maninis, K.-K., Pont-Tuset, J., Leal-Taixe, L., Cremers, D., & Gool, L. Van. (2017). One-Shot Video Object Segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5320–5329. https://doi.org/10.1109/CVPR.2017.565

Cai, J., Denby, B., Roussel, P., Dreyfus, G., & Crevier-Buchman, L. (2011). Recognition and real-time performances of a lightweight ultrasound-based silent speech interface employing a language model. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (August), 1005–1008.

Campbell, F., Gick, B., Wilson, I., & Vatikiotis-Bateson, E. (2010). Spatial and temporal properties of gestures in North American English /r/. *Language and Speech*, *53*(1), 49–69. https://doi.org/10.1177/0023830909351209

Campbell, F. M. (2004). *The gestural organization of North American English/r: A study of timing and magnitude* (University of British Columbia). https://doi.org/10.14288/1.0091851

Chaurasia, A., & Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation. *2017 IEEE Visual Communications and Image Processing (VCIP)*, 1–4.

Chen, K., & Tao, W. (2018). Once for All: A Two-Flow Convolutional Neural Network for Visual Tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(12), 3377–3386. https://doi.org/10.1109/TCSVT.2017.2757061

Chen, L.-C. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 834–848. https://doi.org/10.1109/TPAMI.2017.2699184

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. *ArXiv Preprint ArXiv:1412.7062*.

Chen, L.-C., Papandreou, G., Schroff, F., & Adam, H. (2017). *Rethinking Atrous Convolution for Semantic Image Segmentation*. Retrieved from https://arxiv.org/abs/1706.05587

Chen, L.-C., Yang, Y., Wang, J., Xu, W., & Yuille, A. L. (2016). Attention to Scale: Scale-Aware Semantic Image Segmentation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, *2016-Decem*, 3640–3649. https://doi.org/10.1109/CVPR.2016.396

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*: *Vol. 11211 LNCS* (pp. 833–851). https://doi.org/10.1007/978-3-030-01234-2_49

Chollet, F., & others. (2015). Keras: Deep learning library for theano and tensorflow. *URL: Https://Keras. Io/K*, *7*, 8.

Cootes, T. F., Edwards, G. J., & Taylor, C. J. (2001). Active Appearance Models æ. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *23*(6), 681–685.

Davidson, L. (2006). Comparing tongue shapes from ultrasound imaging using smoothing spline analysis of variance. *The Journal of the Acoustical Society of America*, *120*(1),

407–415. https://doi.org/10.1121/1.2205133

Denby, B., Schultz, T., Honda, K., Hueber, T., Gilbert, J. M. J. M., & Brumberg, J. S. J. S. (2010). Silent speech interfaces. *Speech Communication*, *52*(4), 270–287. https://doi.org/10.1016/j.specom.2009.08.002

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Deng, L. (2014). Deep Learning: Methods and Applications. *Foundations and Trends® in Signal Processing*, *7*(3–4), 197–387. https://doi.org/10.1561/2000000039

Denil, M., Bazzani, L., Larochelle, H., & de Freitas, N. (2012). Learning where to attend with deep architectures for image tracking. *Neural Computation*, *24*(8), 2151–2184. https://doi.org/10.1162/NECO_a_00312

Derrick, D., Carignan, C., Chen, W., Shujau, M., & Best, C. T. (2018). Three-dimensional printable ultrasound transducer stabilization system. *The Journal of the Acoustical Society of America*, *144*(5), EL392--EL398.

Drucker, H., Wu, D., & Vapnik, V. N. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, *10*(5), 1048–1054. https://doi.org/10.1109/72.788645

enunciate UBC. (2019). Retrieved from https://enunciate.arts.ubc.ca/

Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, *111*(1), 98–136.

Everingham, M., & Winn, J. (2012). *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Development Kit*.

Fabre, D., Hueber, T., Bocquelet, F., & Badin, P. (2015). Tongue tracking in ultrasound images using eigentongue decomposition and artificial neural networks. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, *2015-Janua*(2), 2410–2414.

Fabre, D., Hueber, T., Girin, L., Alameda-Pineda, X., & Badin, P. (2017). Automatic animation of an articulatory tongue model from ultrasound images of the vocal tract. *Speech Communication*, *93*, 63–75.

Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., … Ronneberger, O. (2019). Author Correction: U-NET: deep learning for cell counting, detection, and morphometry. *Nature Methods*, *16*(4), 351. https://doi.org/10.1038/s41592-018-0261-2

Fasel, I., & Berry, J. (2010). Deep belief networks for real-time extraction of tongue contours from ultrasound during speech. *Pattern Recognition (ICPR), 2010 20th*

*International Conference On*, 1493–1496. https://doi.org/10.1109/ICPR.2010.369

Fei-Fei, L., Fergus, R., & Perona, P. (2006). One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(4), 594–611. https://doi.org/10.1109/TPAMI.2006.79

Fern, G., Nebehay, G., Pflugfelder, R., Gupta, A., & Bibi, A. (2015). The Visual Object Tracking VOT2015 Challenge Results. *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, 564–586. https://doi.org/10.1109/ICCVW.2015.79

Fischer, A., & Igel, C. (2012). An Introduction to Restricted Boltzmann Machines. In *Progress in Brain Research* (Vol. 207, pp. 14–36). https://doi.org/10.1007/978-3-642-33275-3_2

Fu, J., Liu, J., Wang, Y., Zhou, J., Wang, C., & Lu, H. (2019). Stacked deconvolutional network for semantic segmentation. *IEEE Transactions on Image Processing*.

Gai, S., Zhang, B., Yang, C., & Yu, L. (2018). Speckle noise reduction in medical ultrasound image using monogenic wavelet and Laplace mixture distribution. *Digital Signal Processing: A Review Journal*, *72*, 192–207. https://doi.org/10.1016/j.dsp.2017.10.006

Gan, Q., Guo, Q., Zhang, Z., & Cho, K. (2015). *First Step toward Model-Free, Anonymous Object Tracking with Recurrent Neural Networks*. 1–13. Retrieved from http://arxiv.org/abs/1511.06425

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., & Garcia-Rodriguez, J. (2017). A review on deep learning techniques applied to semantic segmentation. *ArXiv Preprint ArXiv:1704.06857*.

Géron, A. (2017). *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, Inc.

Ghiasi, G., & Fowlkes, C. C. (2016). Laplacian pyramid reconstruction and refinement for semantic segmentation. *European Conference on Computer Vision*, 519–534.

Ghrenassia, S., Ménard, L., & Laporte, C. (2014). Interactive segmentation of tongue contours in ultrasound video sequences using quality maps. *Medical Imaging 2014: Image Processing*, *9034*, 903440.

Gick, B. (2002). The use of ultrasound for linguistic phonetic fieldwork. *Journal of the International Phonetic Association*, *32*(2), 113–121. https://doi.org/10.1017/S0025100302001007

Gick, B., Bernhardt, B. M., Bacsfalvi, P., & Wilson, I. (2008). Ultrasound imaging applications in second language acquisition. In *Phonology and second language acquisition* (Vol. 36, pp. 309–322). https://doi.org/10.1075/sibil.36.15gic

Gick, B., Bird, S., & Wilson, I. (2005). Techniques for field application of lingual

ultrasound imaging. *Clinical Linguistics and Phonetics*, *19*(6–7), 503–514. https://doi.org/10.1080/02699200500113590

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning*. MIT press Cambridge.

Goodfellow, I., Pouget-Abadie, J., & Mirza, M. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 2672–2680.

Gori, M., & Scarselli, F. (1998). Are multilayer perceptrons adequate for pattern recognition and verification? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(11), 1121–1132. https://doi.org/10.1109/34.730549

Graves, A., Mohamed, A. R., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, (6), 6645–6649. https://doi.org/10.1109/ICASSP.2013.6638947

Guo, Y., Liu, Y., Georgiou, T., & Lew, M. S. (2018). A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval*, *7*(2), 87–93. https://doi.org/10.1007/s13735-017-0141-z

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, *187*, 27–48. https://doi.org/10.1016/j.neucom.2015.09.116

Hahn-powell, G. V., Archangeli, D., Berry, J., & Fasel, I. (2014). AutoTrace: An automatic system for tracing tongue contours. *The Journal of the Acoustical Society of America*, *136*(4), 2104. https://doi.org/10.1121/1.4899570

Hamaguchi, R., Fujita, A., Nemoto, K., Imaizumi, T., & Hikosaka, S. (2018). Effective Use of Dilated Convolutions for Segmenting Small Object Instances in Remote Sensing Imagery. *Proceedings - 2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018*, *2018-Janua*, 1442–1450. https://doi.org/10.1109/WACV.2018.00162

Hamed Mozaffari, M., & Lee, W.-S. (2019). Domain adaptation for ultrasound tongue contour extraction using transfer learning: A deep learning approach. *The Journal of the Acoustical Society of America*, *146*(5), EL431–EL437. https://doi.org/10.1121/1.5133665

Hamed Mozaffari, M., Wen, S., Wang, N., Lee, W. S., Mozaffari, M., Wen, S., … Lee., W. (2019). Real-time Automatic Tongue Contour Tracking in Ultrasound Video for Guided Pronunciation Training. *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, *1*, 302–309. https://doi.org/10.5220/0007523503020309

Hansson, P. (2002). Fracture Analysis of Adhesive Joints Using The Finite Element

Method. *Lund Institute of Technology*, *60*(February), 84–90. https://doi.org/10.1145/3065386

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. *Proceedings of the IEEE International Conference on Computer Vision*, 2961–2969.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Hinton, G. E. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, *313*(5786), 504–507. https://doi.org/10.1126/science.1127647

Hinton, Geoffrey E., Osindero, S., & Teh, Y.-W. (2006). A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, *18*(7), 1527–1554. https://doi.org/10.1162/neco.2006.18.7.1527

Hinton, Geoffrey E., & Sejnowski, T. J. (1986). Learning and releaming in boltzmann machines. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, *1*, 282–317.

Hinton, Geoffrey E., & Zemel, R. S. (1994). Autoencoders, minimum description length and Helmholtz free energy. *Advances in Neural Information Processing Systems*, 3–10.

Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., & Liu, W. (2019). Ccnet: Criss-cross attention for semantic segmentation. *Proceedings of the IEEE International Conference on Computer Vision*, 603–612. Retrieved from http://arxiv.org/abs/1811.11721

Hueber, T. (2013). Ultraspeech-player: Intuitive visualization of ultrasound articulatory data for speech therapy and pronunciation training. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (January 2013), 752–753.

Insana, M. F., Wagner, R. F., Garra, B. S., Brown, D. G., & Shawker, T. H. (1985). Analysis Of Ultrasound Image Texture Via Generalized Rician Statistics. In H. H. Arsenault (Ed.), *Intl Conf on Speckle* (Vol. 0556, p. 153). https://doi.org/10.1117/12.949535

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv Preprint ArXiv:1502.03167*.

Iskarous, K. (2005). Detecting the edge of the tongue: A tutorial. *Clinical Linguistics and Phonetics*, *19*(6–7), 555–565. https://doi.org/10.1080/02699200500113871

Jaumard-Hakoun, A., Xu, K., Leboullenger, C., Roussel-Ragot, P., & Denby, B. (2016). An articulatory-based singing voice synthesis using tongue and lips imaging. *Proceedings of the Annual Conference of the International Speech Communication*

*Association, INTERSPEECH*, *08-12-Sept*, 1467–1471. https://doi.org/10.21437/Interspeech.2016-385

Jaumard-Hakoun, A., Xu, K., & others. (2015). Tongue contour extraction from ultrasound images based on deep neural network. *The International Congress of Phonetic Sciences*.

Jaumard-Hakoun, A., Xu, K., Roussel-Ragot, P., Dreyfus, G., & Denby, B. (2016). Tongue contour extraction from ultrasound images based on deep neural network. *Proceedings of the 18th International Congress of Phonetic Sciences (ICPhS 2015)*. Retrieved from http://arxiv.org/abs/1605.05912

Jaumard-Hakoun, A., Xu, K., Roussel-ragot, P., & Stone, M. L. (2015). Tongue Contour Extraction From Ultrasound Images. *Proceedings of the 18th International Congress of Phonetic Sciences (ICPhS 2015)*.

Jégou, S., Drozdzal, M., Vazquez, D., Romero, A., & Bengio, Y. (2017). The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 11–19.

Jiang, F., Grigorev, A., Rho, S., Tian, Z., Fu, Y., Jifara, W., … Liu, S. (2018). Medical image semantic segmentation based on deep learning. *Neural Computing and Applications*, *29*(5), 1257–1265.

Johnson, K. A., Mellesmoen, G. M., Lo, R. Y.-H., & Gick, B. (2018). Prior Pronunciation Knowledge Bootstraps Word Learning. *Frontiers in Communication*, *3*, 1. https://doi.org/10.3389/fcomm.2018.00001

Jouppi, N. P., Young, C., Patil, N., Patterson, D., Agrawal, G., Bajwa, R., … Yoon, D. H. (2017). In-datacenter performance analysis of a tensor processing unit. *Proceedings - International Symposium on Computer Architecture*, *Part F1286*, 1–12. https://doi.org/10.1145/3079856.3080246

Kahou, S. E., Michalski, V., Memisevic, R., Pal, C., & Vincent, P. (2017). RATM: Recurrent Attentive Tracking Model. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, *2017-July*, 1613–1622. https://doi.org/10.1109/CVPRW.2017.206

Kalal, Z., Mikolajczyk, K., & Matas, J. (2012). Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(7), 1409–1422. https://doi.org/10.1109/TPAMI.2011.239

Karimi, E., Ménard, L., & Laporte, C. (2019). Fully-automated tongue detection in ultrasound images. *Computers in Biology and Medicine*, *111*(June), 103335. https://doi.org/10.1016/j.compbiomed.2019.103335

Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models.

*International Journal of Computer Vision*, *1*(4), 321–331. https://doi.org/10.1007/BF00133570

Kelsey, C. A., Woodhouse, R. J., & Minifie, F. D. (1969). Ultrasonic Observations of Coarticulation in the Pharynx. *The Journal of the Acoustical Society of America*, *46*(4B), 1016–1018. https://doi.org/10.1121/1.1911793

Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *ArXiv Preprint ArXiv:1412.6980*. Retrieved from https://arxiv.org/pdf/1412.6980.pdf

Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *ArXiv Preprint ArXiv:1312.6114*.

Kirillov, A., He, K., Girshick, R., Rother, C., & Dollár, P. (2019). Panoptic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9404–9413.

Koch, G., & Koch, G. (2015). *Siamese Neural Network Thesis*.

Kojouharov, S. (n.d.). Neural networks cheat sheet. Retrieved November 5, 2019, from https://becominghuman.ai/cheat-sheets-for-ai-neural-networks-machine-learning-deep-learning-big-data-678c51b4b463

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *2*(6), 1097–1105. https://doi.org/10.1145/3065386

L., T., T., B., G., H., Tang, L., Bressmann, T., & Hamarneh, G. (2012). Tongue contour tracking in dynamic ultrasound via higher-order MRFs and efficient fusion moves. *Medical Image Analysis*, *16*(8), 1503–1520. https://doi.org/10.1016/j.media.2012.07.001

Lai, M. (2015). *Deep Learning for Medical Image Segmentation*.

Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., & Navab, N. (2016). Deeper depth prediction with fully convolutional residual networks. *2016 Fourth International Conference on 3D Vision (3DV)*, 239–248. https://doi.org/10.1109/3DV.2016.32

Laporte, C., & Ménard, L. (2015). Robust tongue tracking in ultrasound images: a multi-hypothesis approach. *Sixteenth Annual Conference of the International Speech Communication Association*.

Laporte, C., & Ménard, L. (2018). Multi-hypothesis tracking of the tongue surface in ultrasound video recordings of normal and impaired speech. *Medical Image Analysis*, *44*, 98–114. https://doi.org/10.1016/j.media.2017.12.003

Lawson, E., Stuart-Smith, J. (2019). *Seeing Speech: an articulatory web resource for the study of phonetics*. Retrieved from https://seeingspeech.ac.uk

Lawson, E., Stuart-Smith, J., Scobbie, J. M., Nakai, S., Beavan, D., Edmonds, F., … others. (2015). *Seeing Speech: an articulatory web resource for the study of phonetics*

*[website]*.

Lee, S. A. S., Wrench, A., & Sancibrian, S. (2015). How To Get Started With Ultrasound Technology for Treatment of Speech Sound Disorders. *SIG 5 Perspectives on Speech Science and Orofacial Disorders*, *25*(2), 66–80.

Li, M., Kambhamettu, C., & Stone, M. (2005). Automatic contour tracking in ultrasound images. *Clinical Linguistics & Phonetics*, *19*(6–7), 545–554. https://doi.org/10.1080/02699200500113616

Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., & Van Den Hengel, A. (2013). A survey of appearance models in visual object tracking. In *ACM Transactions on Intelligent Systems and Technology* (Vol. 4). https://doi.org/10.1145/2508037.2508039

Lin, G., Milan, A., Shen, C., & Reid, I. (2017). RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* (Vol. 2017-Janua). https://doi.org/10.1109/CVPR.2017.549

Lin, T.-Y. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, *2017-Janua*, 2117–2125. https://doi.org/10.1109/CVPR.2017.106

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., … Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, *42*(1995), 60–88. https://doi.org/10.1016/j.media.2017.07.005

Liu, S., Wang, Y., Yang, X., Lei, B., Liu, L., Li, S. X., … Wang, T. (2019). Deep Learning in Medical Ultrasound Analysis: A Review. *Engineering*, *5*(2), 261–275. https://doi.org/10.1016/j.eng.2018.11.020

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, *234*(December 2016), 11–26. https://doi.org/10.1016/j.neucom.2016.12.038

Liu, X., Deng, Z., & Yang, Y. (2019). Recent progress in semantic image segmentation. *Artificial Intelligence Review*, *52*(2), 1089–1106.

Liu, X. Y., Wu, J., & Zhou, Z. H. (2006). Exploratory under-sampling for class-imbalance learning. *Proceedings - IEEE International Conference on Data Mining, ICDM*, *39*(2), 965–969. https://doi.org/10.1109/ICDM.2006.68

Lo, S. C. B., Lou, S. L. A., Chien, M. V., & Mun, S. K. (1995). Artificial Convolution Neural Network Techniques and Applications for Lung Nodule Detection. *IEEE Transactions on Medical Imaging*, *14*(4), 711–718. https://doi.org/10.1109/42.476112

Long, J., Shelhamer, E., Darrell, T., Long, J., Darrell, T., Shelhamer, E., … Darrell, T.

(2015). Fully convolutional networks for semantic segmentation. *Proc. of the ICCVPR*, *39*(4), 3431–3440. https://doi.org/10.1109/TPAMI.2016.2572683

Loosvelt, M., Villard, P.-F., & Berger, M.-O. (2014). Using a biomechanical model for tongue tracking in ultrasound images. *Biomedical Simulation*, *8789*, 67–75. Retrieved from http://link.springer.com/chapter/10.1007/978-3-319-12057-7_8

Luchies, A. C., & Byram, B. C. (2018). Deep Neural Networks for Ultrasound Beamforming. *IEEE Transactions on Medical Imaging*, *37*(9), 2010–2021. https://doi.org/10.1109/TMI.2018.2809641

Makhzani, A., & Frey, B. (2013). k-Sparse Autoencoders. *The Journal of Animal Ecology*, *48*(2), 427. Retrieved from http://arxiv.org/abs/1312.5663

Mamoshina, P., Vieira, A., Putin, E., & Zhavoronkov, A. (2016). Applications of deep learning in biomedicine. *Molecular Pharmaceutics*, *13*(5), 1445–1454.

Martínková, N., Nová, P., Sablina, O. V., Graphodatsky, A. S., & Zima, J. (2004). Karyotypic relationships of the Tatra vole (Microtus tatricus). *Folia Zoologica*, *53*(3), 279–284. https://doi.org/10.1007/s13398-014-0173-7.2

Matan, O., Burges, J. C., LeCun, Y., & Denker, J. S. (1992). Multi-digit recognition using a space displacement neural network. *Proc. NIPS*, 488–495. Retrieved from https://pdfs.semanticscholar.org/464e/8d981df7f326c3af6e9d7bd627f83e438816.pdf

medicaldecathlon. (2019). Retrieved from http://medicaldecathlon.com/results.html

Ménard, L., & Noiray, A. (2011). The development of lingual gestures in speech: Experimental approach to language development. *Faits de Langues*, *37*, 189.

Milletari, F., Navab, N., & Ahmadi, S.-A. A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. *2016 Fourth International Conference on 3D Vision (3DV)*, 565–571. https://doi.org/10.1109/3DV.2016.79

Moeskops, P., Viergever, M. A., Mendrik, A. M., De Vries, L. S., Benders, M. J. N. L., & Isgum, I. (2016). Automatic Segmentation of MR Brain Images with a Convolutional Neural Network. *IEEE Transactions on Medical Imaging*, *35*(5), 1252–1261. https://doi.org/10.1109/TMI.2016.2548501

Mozaffari, M. Hamed, Guan, S., Wen, S., Wang, N., & Lee, W.-S. (2018). Guided Learning of Pronunciation by Visualizing Tongue Articulation in Ultrasound Image Sequences. *2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 1–5. https://doi.org/10.1109/CIVEMSA.2018.8440000

Mozaffari, M. Hamed, & Lee, W.-S. (2019a). *Transfer Learning for Ultrasound Tongue Contour Extraction with Different Domains*. Retrieved from http://arxiv.org/abs/1906.04301

Mozaffari, M. Hamed, Lee, W.-S., Sankoff, D., & Lee, W.-S. (2019). BowNet: Dilated

Convolution Neural Network for Ultrasound Tongue Contour Extraction. *The Journal of the Acoustical Society of America*, *146*(4), 3–6. https://doi.org/10.1121/1.5137212

Mozaffari, M. Hamed, & Lee, W. (2019b). *Real-time Ultrasound Tongue Contour Extraction using Dilated Convolution Neural Network*.

Mozaffari, M.H., & Lee, W.-S. (2017). Convergent heterogeneous particle swarm optimisation algorithm for multilevel image thresholding segmentation. *IET Image Processing*, *11*(8), 605–619. https://doi.org/10.1049/iet-ipr.2016.0489

Mozaffari, M Hamed, & Lee, W. (2019). *Real-time Ultrasound-enhanced Multimodal Imaging of Tongue using 3D Printable Stabilizer System: A Deep Learning Approach*. Retrieved from http://arxiv.org/abs/1911.09840

Mozaffari, M Hamed, Ratul, M. A. R., & Lee, W. (2019). *IrisNet: Deep Learning for Automatic and Real-time Tongue Contour Tracking in Ultrasound Video Data using Peripheral Vision*. 1–12. Retrieved from http://arxiv.org/abs/1911.03972

Mozaffari, Mohammad Hamed, Abdy, H., & Zahiri, S. H. (2016). IPO: An inclined planes system optimization algorithm. *Computing and Informatics*, *35*(1), 222–240.

Mozaffari, Mohammad Hamed, & Lee, W.-S. (2017). Freehand 3-D Ultrasound Imaging: A Systematic Review. *Ultrasound in Medicine & Biology*, *43*(10), 2099–2124. https://doi.org/10.1016/j.ultrasmedbio.2017.06.009

Mozaffari, Mohammad Hamed, & Lee, W. (2016). *3D Ultrasound image segmentation: A Survey*. Retrieved from http://arxiv.org/abs/1611.09811

Mozaffari, Mohammad Hamed, Lee, W., Mozaffari, M. H., Science, C., & Studies, E. (2019). *Real-time and Fully Automatic Ultrasound Multimodal Visual Biofeedback for Second Language Teaching and Learning: A Deep Learning Approach*.

Nekrasov, V., Shen, C., & Reid, I. (2018). Light-Weight RefineNet for Real-Time Semantic Segmentation. *ArXiv Preprint ArXiv:1810.03272*, 1–19. Retrieved from http://arxiv.org/abs/1810.03272

Noguchi, M., Yamane, N., Tsuda, A., Kazama, M., Kim, B., & Gick, B. (2015). Towards protocols for L2 pronunciation training using ultrasound imaging. *Poster Presentation at the 7th Annual Pronunciation in Second Language Learning and Teaching (PSLLT) Conference. Dallas, TX*, 2015.

Noh, H., Hong, S., & Han, B. (2015). Learning deconvolution network for semantic segmentation. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 1520–1528. https://doi.org/10.1109/ICCV.2015.178

Och, F. J., & Ney, H. (2001). Discriminative training and maximum entropy models for statistical machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*, (July), 295.

https://doi.org/10.3115/1073083.1073133

Odena, A., Dumoulin, V., & Olah, C. (2016). Deconvolution and Checkerboard Artifacts. *Distill*, *1*(10), e3. https://doi.org/10.23915/distill.00003

Osman, A., & Kaftandjian, V. (2017). Characterization of speckle noise in three dimensional ultrasound data of material components. *AIMS Materials Science*, *4*(4), 920–938. https://doi.org/10.3934/matersci.2017.4.920

Pan, S., & Yang, Q. (2010). *A survey on transfer learning. IEEE Transaction on Knowledge Discovery and Data Engineering, 22 (10).* Retrieved from https://ieeexplore.ieee.org/abstract/document/5288526

paperswithcode. (2019). Retrieved from https://paperswithcode.com/task/lesion-segmentation

Paszke, A., Chaurasia, A., Kim, S., & Culurciello, E. (2016). *ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation*. 1–10. Retrieved from http://arxiv.org/abs/1606.02147

Poudel, R. P. K., Liwicki, S., & Cipolla, R. (2019). Fast-SCNN: fast semantic segmentation network. *ArXiv Preprint ArXiv:1902.04502*. Retrieved from http://arxiv.org/abs/1902.04502

Preston, J. L., McAllister Byun, T., Boyce, S. E., Hamilton, S., Tiede, M., Phillips, E., … Whalen, D. H. (2017). Ultrasound Images of the Tongue: A Tutorial for Assessment and Remediation of Speech Sound Errors. *Journal of Visualized Experiments*, *2017*(119), 1–10. https://doi.org/10.3791/55123

Prince, Jerry L., Links, J. M. (2015). *Medical imaging signals and systems* (8th ed.). Pearson.

Qi, Y., Zhang, S., Qin, L., Yao, H., Huang, Q., Lim, J., & Yang, M. H. (2016). Hedged Deep Tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 4303–4311. https://doi.org/10.1109/CVPR.2016.466

Rabbani, H., Vafadust, M., Abolmaesumi, P., & Gazor, S. (2008). Speckle Noise Reduction of Medical Ultrasound Images in Complex Wavelet Domain Using Mixture Priors. *IEEE Transactions on Biomedical Engineering*, *55*(9), 2152–2160. https://doi.org/10.1109/TBME.2008.923140

Reynolds, D. A., Quatieri, T. F., & Dunn, R. B. (2000). Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing: A Review Journal*, *10*(1), 19–41. https://doi.org/10.1006/dspr.1999.0361

Ribeiro, M. S., Eshky, A., Richmond, K., & Renals, S. (2019). Ultrasound tongue imaging for diarization and alignment of child speech therapy sessions. *ArXiv Preprint ArXiv:1907.00818*. Retrieved from http://arxiv.org/abs/1907.00818

Richmond, K., & Renals, S. (2012). Ultrax: An animated midsagittal vocal tract display for speech therapy. *Thirteenth Annual Conference of the International Speech Communication Association*.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, (1), 833–840.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-NET: Convolutional Networks for Biomedical Image Segmentation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 9351, pp. 234–241). https://doi.org/10.1007/978-3-319-24574-4_28

Rosenholtz, R. (2016). Capabilities and limitations of peripheral vision. *Annual Review of Vision Science*, *2*, 437–457.

Roussos, A., Katsamanis, A., & Maragos, P. (2009). Tongue tracking in Ultrasound images with Active Appearance Models. *2009 16th IEEE International Conference on Image Processing (ICIP)*, 1733–1736. https://doi.org/10.1109/ICIP.2009.5414520

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *ArXiv Preprint ArXiv:1609.04747*, 1–14. Retrieved from http://arxiv.org/abs/1609.04747

Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic Routing Between Capsules. *Advances in Neural Information Processing Systems*, 3856–3866.

Sak, H., Senior, A., & Françoise, B. (2014). Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling. *Fifteenth Annual Conference of the International Speech Communication Association*.

Salakhutdinov, R., & Larochelle, H. (2010). Efficient learning of Deep Boltzmann Machines. *Journal of Machine Learning Research*, *9*, 693–700.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *07-12-June*, 815–823. https://doi.org/10.1109/CVPR.2015.7298682

Scobbie, J. M., Stuart-Smith, J., & Lawson, E. (2011). Looking variation and change in the mouth: developing the sociolinguistic potential of Ultrasound Tongue Imaging. *Developing the Sociolinguistic Potential of Ultrasound Tongue Imaging*, 1–23.

Scobbie, J. M., Wrench, A. A., & van der Linden, M. (2008). Head-Probe stabilisation in ultrasound tongue imaging using a headset to permit natural head movement. *Proceedings of the 8th International Seminar on Speech Production*.

Sheikh Hassani, M., & Green, J. R. (2019). A semi-supervised machine learning framework for microRNA classification. *Human Genomics*, *13*(Suppl 1), 43.

https://doi.org/10.1186/s40246-019-0221-7

Shen, D., Wu, G., & Suk, H.-I. (2017). Deep Learning in Medical Image Analysis. *Annual Review of Biomedical Engineering*, *19*(1), 221–248. https://doi.org/10.1146/annurev-bioeng-071516-044442

Siam, M., Gamal, M., Abdel-Razek, M., Yogamani, S., & Jagersand, M. (2018). Rtseg: Real-time semantic segmentation comparative study. *2018 25th IEEE International Conference on Image Processing (ICIP)*, (2), 1603–1607. https://doi.org/10.1109/ICIP.2018.8451495

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., … Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489. https://doi.org/10.1038/nature16961

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, *abs/1409.1*. Retrieved from http://arxiv.org/abs/1409.1556

Slud, E., Stone, M., Smith, P. J., & Goldstein Jr., M. (2002). Principal Components Representation of the Two-Dimensional Coronal Tongue Surface. *Phonetica*, *59*(2–3), 108–133. https://doi.org/10.1159/000066066

Smeulders, A. W. M., Chu, D. M., Cucchiara, R., Calderara, S., Dehghan, A., & Shah, M. (2014). Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(7), 1442–1468. https://doi.org/10.1109/TPAMI.2013.230

Socher, R., Lin, C. C. Y., Ng, A. Y., & Manning, C. D. (2011). Parsing natural scenes and natural language with recursive neural networks. *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 129–136.

Socher, R., Lin, C. C. Y., Ng, A. Y., Manning, C. D., Sutton, R. S., & G. Barto, A. (1998). REINFORCEMENT LEARNING: AN INTRODUCTION. *MIT Press, Cambridge, MA, USA*, *17*, 229–235.

Sonies, B. C., Shawker, T. H., Hall, T. E., Gerber, L. H., & Leighton, S. B. (1981). Ultrasonic visualization of tongue motion during speech. *The Journal of the Acoustical Society of America*, *70*(3), 683–686. https://doi.org/10.1121/1.386930

Spreafico, L., Pucher, M., & Matosova, A. (2018). UltraFit: A speaker-friendly headset for ultrasound recordings in speech science. *International Speech Communication Association*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Physics Letters B*, *15*, 1929–1958.

Stollenga, M. F., Byeon, W., Liwicki, M., & Schmidhuber, J. (2015). Parallel multi-

dimensional LSTM, with application to fast biomedical volumetric image segmentation. *Advances in Neural Information Processing Systems*, *2015-Janua*, 2998–3006.

Stoltzfus, J. C. (2011). Logistic regression: A brief primer. *Academic Emergency Medicine*, *18*(10), 1099–1104. https://doi.org/10.1111/j.1553-2712.2011.01185.x

Stone, M. (2005). A guide to analysing tongue motion from ultrasound images. *Clinical Linguistics & Phonetics*, *19*(6–7), 455–501. https://doi.org/10.1080/02699200500113558

Sudheer Kumar, E., & Shoba Bindu, C. (2019). Medical Image Analysis Using Deep Learning: A Systematic Literature Review. *Communications in Computer and Information Science*, *985*, 81–97. https://doi.org/10.1007/978-981-13-8300-7_8

Sun, M., Zhang, X., Van hamme, H., & Zheng, T. F. (2016). Unseen Noise Estimation Using Separable Deep Auto Encoder for Speech Enhancement. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *24*(1), 93–104. https://doi.org/10.1109/TASLP.2015.2498101

Svensson, A., & Schön, T. B. (2017). A flexible state–space model for learning nonlinear dynamical systems. *Automatica*, *80*, 189–199. https://doi.org/10.1016/j.automatica.2017.02.030

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 2818–2826. https://doi.org/10.1109/CVPR.2016.308

Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). DeepFace: Closing the Gap to Human-Level Performance in Face Verification. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 1701–1708. https://doi.org/10.1109/CVPR.2014.220

Takikawa, T., Acuna, D., Jampani, V., & Fidler, S. (2019). Gated-scnn: Gated shape cnns for semantic segmentation. *Proceedings of the IEEE International Conference on Computer Vision*, 5229–5238. Retrieved from http://arxiv.org/abs/1907.05740

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. *International Conference on Artificial Neural Networks*, 270–279.

Tang, F., Brennan, S., Zhao, Q., & Tao, H. (2007). *Co-Tracking Using Semi-Supervised Support Vector Machines*.

Tang, L., & Hamarneh, G. (2010). Graph-based tracking of the tongue contour in ultrasound sequences with adaptive temporal regularization. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010*, 154–161. https://doi.org/10.1109/CVPRW.2010.5543597

Tao, R., Gavves, E., & Smeulders, A. W. M. (2016). Siamese Instance Search for Tracking. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1420–1429. https://doi.org/10.1109/CVPR.2016.158

Tateishi, M., & Winters, S. (2013). Does ultrasound training lead to improved perception of a non-native sound contrast? Evidence from Japanese learners of English. *Proceedings of the 2013 Annual Conference of the Canadian Linguistic Association*, 1–15.

Thijssen, J. M. (2003). Ultrasonic speckle formation, analysis and processing applied to tissue characterization. *Pattern Recognition Letters*, *24*(4–5), 659–675. https://doi.org/10.1016/S0167-8655(02)00173-3

Vincent, P., & Larochelle, H. (2008). Extracting and Composing Robust Features with Denoising.pdf. *Appearing in Proceedings of the 25th International Confer- Ence on Machine Learning*, 1096–1103.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal OfMachine Learning Research 11*, *11*, 3371–3408.

Vinet, L., & Zhedanov, A. (2010). Rapid Object Detection using a Boosted Cascade of Simple Features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, *1*(C), I-511-I–518. https://doi.org/10.1088/1751-8113/44/8/085201

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., & Wierstra, D. (2016). Matching networks for one shot learning. *Advances in Neural Information Processing Systems*, (Nips), 3637–3645.

Viola, P., & Jones, M. J. (2004). Robust Real-Time Face Detection. *International Journal of Computer Vision*, *57*(2), 137–154. https://doi.org/10.1023/B:VISI.0000013087.49260.fb

Wang, F. Y., Zhang, J. J., Zheng, X., Wang, X., Yuan, Y., Dai, X., … Yang, L. (2016). Where does AlphaGo go: From church-turing thesis to AlphaGo thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, *3*(2), 113–120. https://doi.org/10.1109/JAS.2016.7471613

Web. (2019). Sota. Retrieved from https://paperswithcode.com/sota

Wilson, I., Gick, B., O'Brien, M. G., Shea, C., & Archibald, J. (2006). Ultrasound Technology and Second Language Acquisition Research Ian. *Proceedings of the 8th Generative Approaches to Second Language Acquisition Conference (GASLA 2006)*, 148–152. https://doi.org/doi.org/10.1002/hed.20772

Wrench, A. A., Cleland, J., & Scobbie, J. M. (2011). An ultrasound protocol for comparing tongue contours: upright vs. supine. *Proceedings of 17th ICPhS, Hong Kong*, 2161–

2164.

Wu, Y., & Ji, Q. (2019). Facial Landmark Detection: A Literature Survey. *International Journal of Computer Vision*, *127*(2), 115–142. https://doi.org/10.1007/s11263-018-1097-z

Wu, Z., Wang, X., Gonzalez, J. E., Goldstein, T., & Davis, L. S. (2019). ACE: Adapting to Changing Environments for Semantic Segmentation. *ArXiv Preprint ArXiv:1904.06268*.

Xie, Y., Zhang, Z., Sapkota, M., & Yang, L. (2016). Spatial Clockwork Recurrent Neural Network for Muscle Perimysium Segmentation. In *Medical Image Computing and Computer-Assisted Intervention* (Vol. 1, pp. 185–193). https://doi.org/10.1007/978-3-319-46723-8_22

Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical Evaluation of Rectified Activations in Convolutional Network. *ArXiv Preprint ArXiv:1505.00853*. Retrieved from http://arxiv.org/abs/1505.00853

Xu, J., Li, H., & Zhou, S. (2015). An Overview of Deep Generative Models. *IETE Technical Review*, *32*(2), 131–139. https://doi.org/10.1080/02564602.2014.987328

Xu, K., Gábor Csapó, T., Roussel, P., & Denby, B. (2016). A comparative study on the contour tracking algorithms in ultrasound tongue images with automatic re-initialization. *The Journal of the Acoustical Society of America*, *139*(5), EL154–EL160. https://doi.org/10.1121/1.4951024

Xu, K., Roussel, P., Csapó, T. G., & Denby, B. (2017). Convolutional neural network-based automatic classification of midsagittal tongue gestural targets using B-mode ultrasound images. *The Journal of the Acoustical Society of America*, *141*(6), EL531--EL537.

Xu, K., Yang, Y., Stone, M., Jaumard-Hakoun, A., Leboullenger, C., Dreyfus, G., … Denby, B. (2016). Robust contour tracking in ultrasound tongue image sequences. *Clinical Linguistics & Phonetics*, *30*(3–5), 313–327. https://doi.org/10.3109/02699206.2015.1110714

Xu, Y., Xiao, T., Zhang, J., Yang, K., & Zhang, Z. (2014). Scale-invariant convolutional neural networks. *ArXiv Preprint ArXiv:1411.6369*.

Xue, Y., Xu, T., Zhang, H., Long, L. R., & Huang, X. (2018). SegAN: Adversarial Network with Multi-scale L 1 Loss for Medical Image Segmentation. *Neuroinformatics*, *16*(3–4), 383–392. https://doi.org/10.1007/s12021-018-9377-x

Yamane, N., Abel, J., Allen, B., Burton, S., Kazama, M., Noguchi, M., … Gick, B. (2015). Ultrasound-Integrated Pronunciation Teaching and Learning. *Ultrafest VII, Hong Kong*, (December), 1–4.

Yang, M., Yu, K., Zhang, C., Li, Z., & Yang, K. (2018). Denseaspp for semantic

segmentation in street scenes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3684–3692. https://doi.org/10.1109/CVPR.2018.00388

Yilmaz, A., Javed, O., & Shah, M. (2006a). Object tracking: A survey. *ACM Computing Surveys*, *38*(4), 13. https://doi.org/10.1145/1177352.1177355

Yilmaz, A., Javed, O., & Shah, M. (2006b). Object tracking. *ACM Computing Surveys*, *38*(4), 13-es. https://doi.org/10.1145/1177352.1177355

Yu, F., Koltun, V., & Funkhouser, T. (2017). Dilated residual networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 472–480. Retrieved from http://openaccess.thecvf.com/content_cvpr_2017/papers/Yu_Dilated_Residual_Networks_CVPR_2017_paper.pdf%0Ahttp://openaccess.thecvf.com/content_cvpr_2017/html/Yu_Dilated_Residual_Networks_CVPR_2017_paper.html

Yuen, K.-W., Leung, W.-K., Liu, P., Wong, K.-H., Qian, X., Lo, W.-K., & Meng, H. (2011). Enunciate: An internet-accessible computer-aided pronunciation training system and related user evaluations. *2011 International Conference on Speech Database and Assessments (Oriental COCOSDA)*, 85–90.

Zagoruyko, S., & Komodakis, N. (2015). Learning to compare image patches via convolutional neural networks. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4353–4361. https://doi.org/10.1109/CVPR.2015.7299064

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8689 LNCS*(PART 1), 818–833. https://doi.org/10.1007/978-3-319-10590-1_53

Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. (2010). Deconvolutional networks. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2528–2535. https://doi.org/10.1109/CVPR.2010.5539957

Zhai, M., Chen, L., Mori, G., & Roshtkhari, M. J. (2019). Deep learning of appearance models for online object tracking. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11132 LNCS*, 681–686. https://doi.org/10.1007/978-3-030-11018-5_57

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid Scene Parsing Network. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, *2017-Janua*, 6230–6239. https://doi.org/10.1109/CVPR.2017.660

Zhao, J., Mathieu, M., Goroshin, R., & Lecun, Y. (n.d.). *STACKED WHAT-WHERE*

*AUTO-ENCODERS*. Retrieved from https://arxiv.org/pdf/1506.02351.pdf

Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2018). Object Detection with Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, 1–21. https://doi.org/10.1109/TNNLS.2018.2876865

Zharkova, N. (2013). Using ultrasound to quantify tongue shape and movement characteristics. *The Cleft Palate-Craniofacial Journal*, *50*(1), 76–81.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., … Torr, P. H. S. (2015). Conditional random fields as recurrent neural networks. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 1529–1537. https://doi.org/10.1109/ICCV.2015.179

Zhou, S. K., Greenspan, H., & Shen, D. (2017). *Deep learning for medical image analysis*. Academic Press.

Zhu, J., Styler, W., & Calloway, I. (2019). *A CNN-based tool for automatic tongue contour tracking in ultrasound images*. 1–6. Retrieved from http://arxiv.org/abs/1907.10210

Zhu, X. X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, *5*(4), 8–36.

# Chapter 9 Appendix I

User Study Ethics Approval Certificate and Consent Form:

11/12/2019

Université d'Ottawa
Bureau d'éthique et d'intégrité de la recherche

University of Ottawa
Office of Research Ethics and Integrity

**CERTIFICAT D'APPROBATION ÉTHIQUE | CERTIFICATE OF ETHICS APPROVAL**

| | |
|---|---|
| **Numéro du dossier / Ethics File Number** | H-11-19-5160 |
| **Titre du projet / Project Title** | Ultrasound Technology and Machine Learning for Second Language Pronunciation Training |
| **Type de projet / Project Type** | Thèse de doctorat / Doctoral thesis |
| **Statut du projet / Project Status** | Approuvé / Approved |
| **Date d'approbation (jj/mm/aaaa) / Approval Date (dd/mm/yyyy)** | 11/12/2019 |
| **Date d'expiration (jj/mm/aaaa) / Expiry Date (dd/mm/yyyy)** | 10/12/2020 |

**Équipe de recherche / Research Team**

| Chercheur / Researcher | Affiliation | Role |
|---|---|---|
| M. Hamed MAAREF | École de science informatique et de génie électrique / School of Electrical Engineering and Computer Science | Chercheur Principal / Principal Investigator |
| Wonsook LEE | École de science informatique et de génie électrique / School of Electrical Engineering and Computer Science | Superviseur / Supervisor |

**Conditions spéciales ou commentaires / Special conditions or comments**

550, rue Cumberland, pièce 154
Ottawa (Ontario) K1N 6N5 Canada

550 Cumberland Street, Room 154
Ottawa, Ontario K1N 6N5 Canada

613-562-5387 • 613-562-5338 • ethique@uOttawa.ca / ethics@uOttawa.ca
www.recherche.uottawa.ca/deontologie | www.recherche.uottawa.ca/ethics

Université d'Ottawa
Bureau d'éthique et d'intégrité de la recherche

University of Ottawa
Office of Research Ethics and Integrity

Le Comité d'éthique de la recherche (CÉR) de l'Université d'Ottawa, opérant conformément à l'*Énoncé de politique des Trois conseils* (2014) et toutes autres lois et tous règlements applicables, a examiné et approuvé la demande d'éthique du projet de recherche ci-nommé.

L'approbation est valide pour la durée indiquée plus haut et est sujette aux conditions énumérées dans la section intitulée "Conditions Spéciales ou Commentaires". Le formulaire « Renouvellement ou Fermeture de Projet » doit être complété quatre semaines avant la date d'échéance indiquée ci-haut afin de demander un renouvellement de cette approbation éthique ou afin de fermer le dossier.

Toutes modifications apportées au projet doivent être approuvées par le CÉR avant leur mise en place, sauf si le participant doit être retiré en raison d'un danger immédiat ou s'il s'agit d'un changement ayant trait à des éléments administratifs ou logistiques du projet. Les chercheurs doivent aviser le CÉR dans les plus brefs délais de tout changement pouvant augmenter le niveau de risque aux participants ou pouvant affecter considérablement le déroulement du projet, rapporter tout évènement imprévu ou indésirable et soumettre toute nouvelle information pouvant nuire à la conduite du projet ou à la sécurité des participants.

The University of Ottawa Research Ethics Board, which operates in accordance with the *Tri-Council Policy Statement* (2014) and other applicable laws and regulations, has examined and approved the ethics application for the above-named research project.

Ethics approval is valid for the period indicated above and is subject to the conditions listed in the section entitled "Special Conditions or Comments". The "Renewal/Project Closure" form must be completedfour weeks before the above-referenced expiry date to request a renewal of this ethics approval or closure of the file.

Any changes made to the project must be approved by the REB before being implemented, except when necessary to remove participants from immediate endangerment or when the modification(s) only pertain to administrative or logistical components of the project. Investigators must also promptly alert the REB of any changes that increase the risk to participant(s), any changes that considerably affect the conduct of the project, all unanticipated and harmful events that occur, and new information that may negatively affect the conduct of the project or the safety of the participant(s).

Germain ZONGO
Responsable d'éthique en recherche / Protocol Officer
Pour/For **Daniel LAGAREC** Président(e) du/ Chair of the **Comité d'éthique de la recherche en sciences de la santé et sciences / Health Sciences and Sciences Research Ethics Board**

550, rue Cumberland, pièce 154
Ottawa (Ontario) K1N 6N5 Canada

550 Cumberland Street, Room 154
Ottawa, Ontario K1N 6N5 Canada

613-562-5387 • 613-562-5338 • ethique@uOttawa.ca / ethics@uOttawa.ca
www.recherche.uottawa.ca/deontologie | www.recherche.uottawa.ca/ethics

uOttawa

Université d'Ottawa
Faculté de génie
École de science informatique et de génie électrique

University of Ottawa
Faculty of Engineering
School of Electrical Engineering and Computer Science

613-562-5738
613-562-5664

800 King Edward
Ottawa ON K1N 6N5 Canada
www.uOttawa.ca

# User Study consent form

## Project Title

**Second Language Pronunciation Training using Ultrasound Technology and Artificial Intelligence**

The participial investigator: **Mohammad Hamed Mozaffari,**
School of Electrical Engineering and Computer Science,
University of Ottawa
SITE Building Room 4009F

Supervisor: **Dr. Won-Sook Lee,**
School of Electrical Engineering and Computer Science,
University of Ottawa
CBY Building Room A509

## Purpose of this study

The purpose of this study is to understand how people react to our pronunciation training system and how people use our system for their pronunciation learning and training. Your participation in this study will help us make the system more comfortable to use, improve its performance, and show its ability to other researchers. We aim to know how our system will help people to learn second language pronunciation easier by real-time usage of ultrasound machines and artificial intelligence techniques. You only need to use our device for several minutes and fill out a questionnaire with questions about your experience. This is a doctoral thesis project.

## Freedom to withdraw

We use an ultrasound device and take some personal information, which is used only for this research. You are free to withdraw from our study any time before, during, or after the experiment session. If you decide to withdraw from the study, we will delete all your data from our dataset including video, audio, and forms.

- You can refuse to take part at any time.
- You can take a break at any time.
- You can ask questions at any time.

uOttawa

Université d'Ottawa
Faculté de génie
École de science informatique et de génie électrique

University of Ottawa
Faculty of Engineering
School of Electrical Engineering and Computer Science

## Information we will collect

We will ask you to show us how you use the product. We will watch how you do various tasks, and we will ask you some questions. We will also record the session, and we will take notes to record your comments and actions.

You have these rights:

- The right to withdraw from the project at any time;
- The right to refuse to answer questions without fear of reprisal or ill treatment;
- The right to be informed of how their identities will be protected in the publication of the data;
- The right to be informed of the limits of confidentiality.

## Tasks of the participant

We have three main sections in the experimental session: pre-training, testing the device, and answering to the questions of the survey. Your task is to listen to the pre-training session, we will do a pre-test for the device to introduce the process to you. You should follow the safety and instructions of the investigator. You will answer the questions of the survey. During all this procedure, you have the right to stop the experiment and leave the lab for any reason. The participation time will be less than 30 minutes approximately including watching a pre-training video, demo of the system, and answering to the questions of the survey. This experimental session can have benefits for you such as knowledge about tongue movements and its importance for pronunciation. You can test your mother language using our system to perceive the tongue gestures in that language. Note that Ultrasound device is considered as a non invasive and safe medical technology, and there is no harm for you in this experiment. Ultrasound gel might not be a comfortable substance for some users. You should let us know if you are not comfortable with using ultrasound gel. For the video recording, we record two videos, one from the screen of our system which is a computer monitor. And we will record a video from the whole experiment using another camera.

## Anonymity and Confidentiality

Your data will be kept confidential, meaning that no one else except the principal investigator and his supervisor will have access to the recordings and your responses to the survey. We may publish research reports that include your comments, but the data used in these reports will be anonymous. This means you will not be identifiable.

☎ 613-562-5738
613-562-5664

800 King Edward
Ottawa ON K1N 6N5 Canada
www.uOttawa.ca

uOttawa

Université d'Ottawa
Faculté de génie
École de science informatique et de génie électrique

University of Ottawa
Faculty of Engineering
School of Electrical Engineering and Computer Science

For the protection of the participant identity, the video files will be manipulated by covering the upper side of the participant face in videos. For the data collected, identity of the participant will be removed in reports and publications.

## Data conservation

We will keep your data from this experiment for five years. It means that after the five years, we will delete all your data. However, you can request the removal of your data at anytime.

## Your agreement

To take part in the research, please sign this form showing that you consent to participate in this study conducted by Mr. Mohammad Hamed Mozaffari under the supervision of Professor WonSook Lee, both form the School of Electrical Engineering and Computer Science at the University of Ottawa.

If you have any questions about the study, please contact the principal investigator or his supervisor. Their contact information is available on the first page of this consent form.

If you have any questions regarding the ethical conduct of this study, you may contact the Protocol Officer for Ethics in Research, University of Ottawa, Tabaret Hall, 550 Cumberland Street, Room 154, Ottawa, ON K1N 6N5, Tel.: (613) 562-5387, Email: ethics@uottawa.ca.

I give my consent to the researchers to participate in this project.

There are two copies of the consent form, one of which is mine to keep.

Name of the participant: ----------

---------- ----------

Signature Date

Name of the researcher: ----------

---------- ----------

Signature Date

613-562-5738
613-562-5664

800 King Edward
Ottawa ON K1N 6N5 Canada
www.uOttawa.ca