Feasibility of Transfer Learning for Automated Segmentation of 2D Echocardiograms

Transferring Segmentation Knowledge of Fully Convolutional Neural Networks from Images of Heart's Left Side to Right Side

Artem Chernyshov

Thesis submitted for the degree of
Master in Informatics: Robotics and Intelligent Systems
60 credits

Department of Informatics
Faculty of mathematics and natural sciences

UNIVERSITY OF OSLO

Spring 2021

Feasibility of Transfer Learning for Automated Segmentation of 2D Echocardiograms

Transferring Segmentation Knowledge of Fully Convolutional Neural Networks from Images of Heart's Left Side to Right Side

Artem Chernyshov

Feasibility of Transfer Learning for Automated Segmentation of 2D Echocardiograms

http://www.duo.uio.no/

Printed: Reprosentralen, University of Oslo

Abstract

The thesis explores the use of transfer learning for automated segmentation of 2D echocardiograms in the situation where few labeled data are available for end-to-end training. The main focus is on transfer learning involving data sets of the left heart chambers and the right heart chambers, with a goal of improving segmentation performance on the latter kind of data. A custom version of the U-Net neural network was implemented for automated segmentation and trained on two left heart data sets and one right heart data set. The worst performing models in direct training on right heart data significantly improved through pre-training on left heart data. Their multi-class Dice Score rose by 6% on average, while the score for RV epicardium improved by 16%. Predictions made by the models were also explored, revealing that the left heart chambers and the right heart chambers share certain features in the images that are useful for learning segmentation. It is concluded that transfer learning appears to be a feasible approach for echocardiogram segmentation when there is a lot of data available in the source task data set and far less data in the target task data set. Using transfer learning with the right setup can therefore reduce the amount of right heart ultrasound data that needs to be collected for training AI segmentation models. However, further investigation is required to quantify some observations made throughout the work.

Acknowledgements

I would like to thank my supervisor Ole Jakob Elle for guidance he offered during this work, and for bringing me in contact with bright minds of the Oslo University Hospital. My gratitude also goes to Kristin McLeod, my unofficial supervisor. She provided data, resources and advice that made completion of the thesis possible. Finally, I want to extend thanks to Henrik Brun for sharing some of his invaluable knowledge and experience as a cardiologist.

Abbreviations

LV	Left Ventricle
RV	Right Ventricle
LA	Left Atrium
RA	Right Atrium
A2C	Apical Two-Chamber (View)
A4C	Apical Four-Chamber (View)
AI	Artificial Intelligence
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
FCNN	Fully Convolutional Neural Network

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Medicine is an extremely large and perpetually developing field of science. As more knowledge is discovered, more opportunities arise within the field. At the same time, the already difficult road to becoming a medical professional gets increasingly more arduous. The existing professionals are not spared either, as they are expected to keep up with the pace of progress and become familiar with new technology and procedures. For example, more than 33% of cardiologists reported feeling burned out in 2015, according to *American College of Cardiology* [1]. Still, developments within Artificial Intelligence (AI) may produce medical technologies that both improve the quality of services and reduce the mental burden of medical professionals.

The discipline of medical imaging seems to lend itself particularly well to integration of AI, given the advancements of Deep Learning in image analysis. AI can be used with any imaging modality, for any body part or organ, to discover patterns in the images and help with their interpretation. In echocardiography, the intersection of medical ultrasound imaging and cardiology, image acquisition and assessment are particularly plagued by inter-observer variability, and AI may offer some relief to the problem [2, 3]. As it currently stands, there is a variation in how different cardiologists acquire and interpret images.

Since a variety of factors can affect human performance, intra-observer variability is also a possible issue. AI can be a lot faster than humans in measuring and interpreting images, while incorporating data from multiple experts to, hopefully, "cancel out" some of the bias.

1.1 Research Motivation

Although AI may provide assistance within echocardiography, a lot of obstacles remain. Firstly, the majority of research on automated interpretation of echocardiograms (ultrasound images of the heart) focuses on the left side of the heart, neglecting the right side in the process. It is a common belief that human heart is largely symmetrical, but an article by *Ostenfeld* and *Flachskampf* dispels this notion, at least in the case of the left ventricle (LV) and the right ventricle (RV) chambers [4]. The authors posit that the shape of RV is more complex than that of LV, making it difficult to visualize and measure. Even so, a possibility remains that RV is similar enough to LV to warrant the use of the same methods for automated interpretation.

Secondly, there is an issue of data availability. While medical institutions store a lot of cardiac ultrasound data, it is rarely annotated or labeled. Therefore, it cannot be used to train AI models in a supervised manner. The issue is particularly severe when it comes to data on the right side of the heart, due to a relative lack of research and interest. Fortunately, there are more ways to combat this issue beyond merely continuing the efforts to label and prepare more data. One approach is to generate and make use of synthetic data [5]. Another possible approach is *transfer learning*.

Transfer learning is a family of techniques within AI that allow for more efficient use of small data sets whenever a larger but similar data set is available. There is a lot of available "left heart" data, but very little "right heart" data. Thus, AI algorithms could use big data sets of

the left heart for pre-training and small data sets of the right heart for fine-tuning, thereby performing a transfer of knowledge. This arrangement should, in theory, provide increased performance in automated interpretation of echocardiograms depicting the right side of the heart.

One important subroutine within echocardiogram assessment is segmentation of the heart regions and chambers. It is a necessary task that serves as a stepping stone for calculating the chamber size, volume, mass, and other important metrics. Currently, skilled cardiologists perform this task by hand, but there is a lot of progress on automating the procedure for the left side of the heart. Similar research for the right side of heart is lacking, however.

This thesis therefore focuses on exploring the feasibility of transfer learning in automated segmentation of the right heart chambers and regions. The expectation is that the approach of transfer learning can provide better segmentation performance than simple training of segmentation algorithms on the right heart data. If transfer learning is indeed feasible for automated segmentation for this problem, it will mean that the currently used segmentation models can generalize better and have more value than previously believed. Furthermore, it will potentially spare the medical industry the need to collect and prepare excessive amounts of heart ultrasound data.

1.2 Goal and Research Question

There is a multitude of intelligent algorithms that are applicable for different subtasks that comprise echocardiogram interpretation. Thankfully, when it comes to the subtask of segmentation, it is reasonable to limit oneself to just one type of algorithms: convolutional encoder-decoders and their derivatives. These algorithms are well-suited for image segmentation in general, while one specific architecture, U-Net, has proven its worth in biomedical applications in partic-

ular.

This work attempts to answer the following question: **"Is transfer learning *feasible* for automated segmentation of the right heart chambers in 2D echocardiograms?"** To clarify the term "feasible" in this context, pre-training algorithms on left heart data before fine-tuning them with limited right heart data should produce better segmentation results than simply training the algorithms on a small data set from scratch. If such pre-training leads to worse results instead, then it would mean that *negative transfer* occurred, and transfer learning would be judged as not "feasible" for the problem. If transfer learning is only effective in a specific set of circumstances, then the goal is to identify which circumstances those would be. See Figure 1.1 for an overview of the direct (end-to-end) training approach and the transfer learning approach.

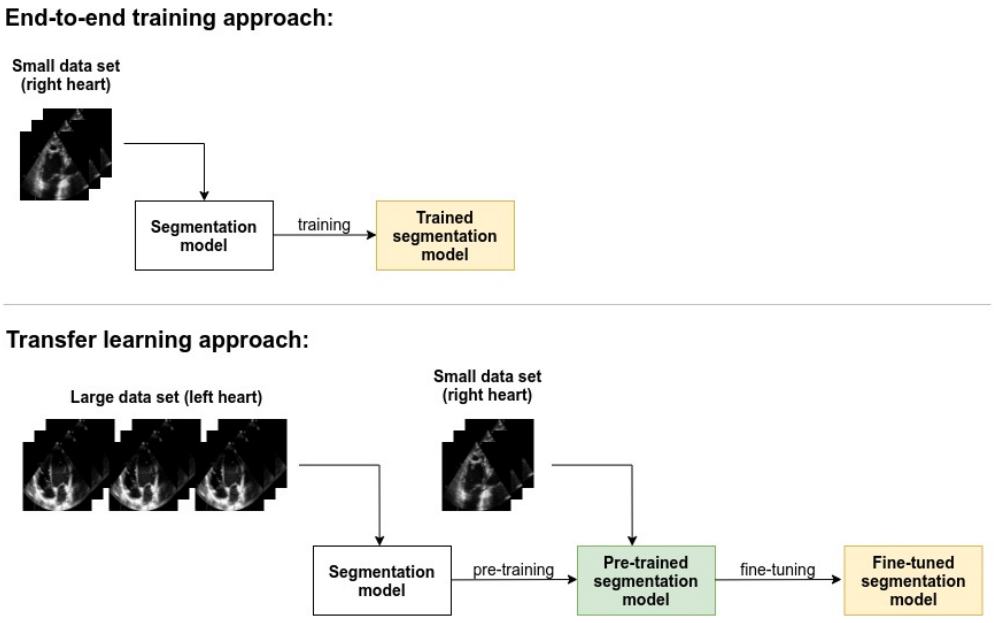


Figure 1.1: The approaches for training the segmentation models. The goal of the thesis is to investigate which approach produces the models that are better at segmenting the right heart echocardiograms.

On the way to answering the main research question, the thesis con-

tributes in a number of related topics. Further attention is brought to the importance of the right side of the heart within cardiology, echocardiography, and image analysis with AI. The work surveys and briefly reviews the current research in AI-assisted echocardiography. Some of the existing echocardiographic data sets and their properties are also discussed.

1.3 Limitations of the Work

Although the thesis attempts to validate the results as extensively as possible, there are inevitable limitations. Transfer learning is only applied to the U-Net algorithm, as it is one of the best artificial neural networks currently available for biomedical segmentation tasks. U-Net is modifiable to a great degree, and testing it under different parameters already takes up a great amount of resources.

Transfer learning is not one specific technique, but a *family* of techniques, or a concept - there are several ways to go about implementing it. The work relies on the simplest implementation, where an algorithm is trained on one data set and then continues training on another. More complex transfer learning techniques are not investigated.

Only one formal metric is used to assess segmentation performance - the Dice Coefficient metric. The obtained data sets simply do not have the information needed to calculate other metrics that would be useful. The metrics that *can* be calculated either strongly correlate with the Dice Coefficient or are not quite appropriate for segmentation tasks.

Further, knowledge transfer involving only three data sets is explored - two of them are the left heart data sets, and the remaining one is the right heart data set. There is no guarantee that the results can be replicated with other data sets.

Finally, the work makes certain assumption about how the efficiency of transfer learning should be evaluated, as there is no estab-

lished method of doing so. The reasoning behind the chosen procedures is explained in the relevant section.

1.4 Note on Implementation

The programs created for the thesis are written in Python programming language (v3.7). The custom implementation of the U-Net algorithm was made with the support of the PyTorch package (v1.7.1). All of the U-Net models were trained on a GeForce RTX3090 Graphics Card.

1.5 Thesis Structure

The thesis consists of six chapters, counting the current one. The next chapter, **Chapter 2** ("Theoretical Background"), introduces the reader to a variety of background topics relevant to the main research question. The topics include medical ultrasound, basic heart anatomy, image analysis with AI, and current achievements within AI-assisted echocardiography.

Chapter 3 ("Overview of Acquired Data") covers the three data sets used for training the segmentation algorithm. The important properties of each data set are discussed, along with their strengths and weaknesses. One of the data sets offers more than segmentation masks, and the features of this data set are analyzed more thoroughly.

Chapter 4 ("Methodology") builds up on the information provided in Chapters 2 and 3 to introduce a range of algorithms and procedures used to answer the main research question. It covers the custom implementation of U-Net created for this work, details on the use of Dice Coefficient metric to assess segmentation performance, explains how trained U-Net models can be applied to produce human-readable

echocardiogram segmentations, and sets up further experiments.

Chapter 5 ("Experiments and Results") is dedicated to experiments required to answer the main research question. First, transfer learning experiments between two left heart data sets are conducted as a trial run of the experimental setup. Next, the setup is optimized, transfer learning experiments between left heart and right heart data sets are performed, and the results are briefly discussed. Finally, a look at the individual predictions made by the trained U-Net models is offered with the purpose to discern additional patterns in the models' behavior.

Chapter 6 ("Discussion and Conclusion") provides an overall assessment of the results. The limitations of the work are then revised with these results in mind. Lastly, possible improvements and extensions to the work are discussed.

Chapter 2

Theoretical Background

This chapter acquaints the reader with relevant background topics for automated segmentation of echocardiograms. It is both a medical and a technical task, requiring at least some understanding of medical imaging, cardiology, and artificial intelligence.

Section 2.1 briefly covers some of the existing medical imaging modalities, focusing on medical ultrasound and diagnostic echocardiography in particular. Section 2.2 is a refresher on the roles of the heart chambers and their current understanding within cardiology. Section 2.3 revises the important concepts within the field of image analysis with AI. Finally, Section 2.4 explores current developments in AI-assisted echocardiography.

2.1 Medical Imaging

Medical Imaging is a sub-discipline with a long history that spans more than 100 years of research. Starting with the discovery of X-rays by Wilhelm Rontgen in 1895, the field of medical imaging has been acquiring a growing set of techniques: X-ray imaging, X-ray based Computed Tomography (CT), Magnetic Resonance Imaging (MRI), and Ul-

trasound (US) imaging.

While all of these technologies have been introduced several decades ago, Medical Imaging is still rapidly developing. New imaging modalities are proposed every few years, and the ever-increasing capabilities of sensors lead to further improvements. Additionally, the emerging disciplines of Computer Vision and Machine Learning can enhance all imaging modalities by providing tools to extract more information from the sensor readings.

As of now, there is no perfect medical imaging technique or modality: each of them has its own advantages and disadvantages when it comes to cost, safety, areas of application, and a myriad other factors. For instance, ultrasound imaging, the focus of this work, distinguishes itself as one of the safest, cost-effective and portable technologies within the field [6]. On the downside, it tends to suffer from lower spatial resolution than CT and MRI.

2.1.1 Ultrasound and Ultrasonic Transducers

The physical definition of *sound* is a vibration that generates an acoustic wave, yet the colloquial definition only refers to acoustic waves in the frequency range between 16 and 20,000 Hz that humans can perceive. Similarly, physicists define *ultrasound* as any sound with frequency above 20,000 Hz, while medics mostly focus on frequencies between 2 and 15 MHz – the typical operational frequency range of a medical *ultrasonic transducer*.

Ultrasonic transducers are devices that enable ultrasound imaging. They come in a large variety of forms to suit different use cases, but they are all built on the same principle that utilizes the piezoelectric properties of ceramic crystals (see Figure 2.1 for an overview of transducer structure). Piezoelectricity is reflexive: materials with this property produce electric current under mechanical pressure and vice versa. Thus, sending electricity through the ceramic crystals forces

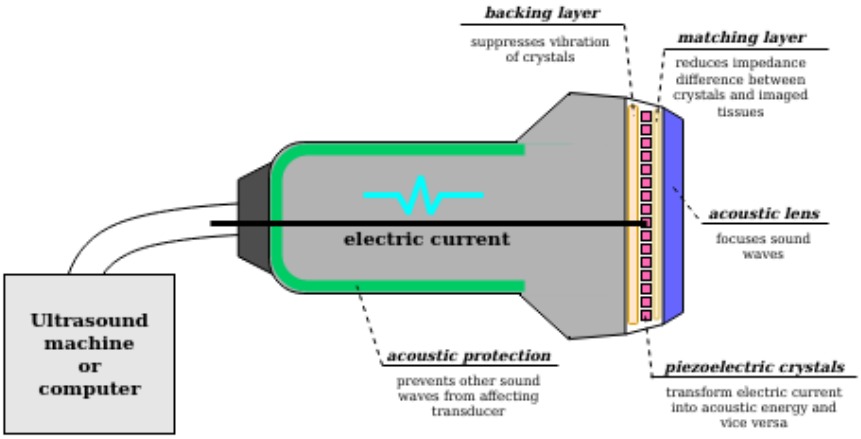


Figure 2.1: The structure of an ultrasonic transducer

them to vibrate and emit ultrasound waves. When the waves are reflected back into the crystals, electric signals are formed instead. The resulting signals are processed and plotted as a sonogram - an ultrasound image. Moreover, measuring the frequency shift can reveal information about the velocity of a moving object via the Doppler Effect. Refer to Figure 2.2 for a more visual explanation of US transducers' operational principles.

Medical applications of this technology are immediately clear: so long as the ultrasound waves can penetrate the skin of a patient, it is possible to form a sonogram of the internal organs and even observe the blood flow. Depending on the physical arrangement of the piezoelectric crystals, a transducer may be capable of real-time 1D (with a single crystal), 2D (with an array of crystals) or even 3D imaging (with a two-dimensional array of crystals or a moving one-dimensional array). Image depth can be controlled by tuning the frequency of the emitted waves, though the crystal arrangement has an effect on it as well.

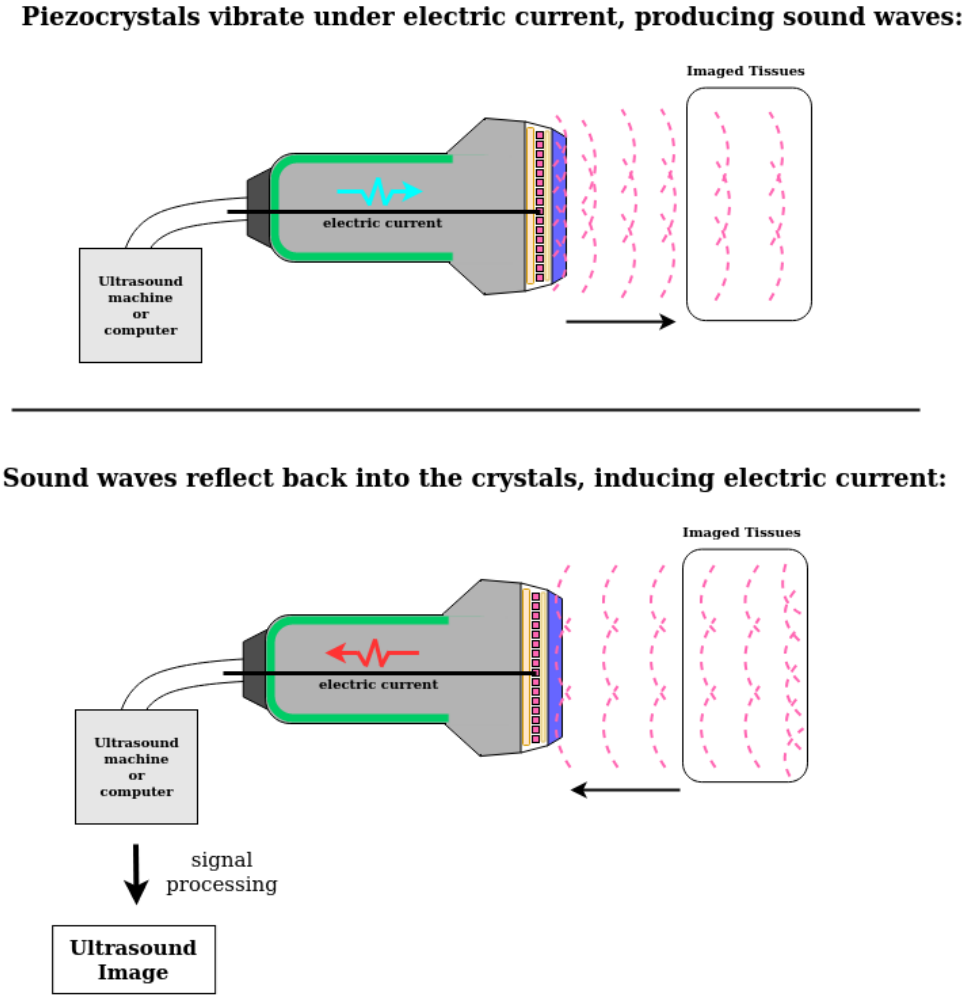


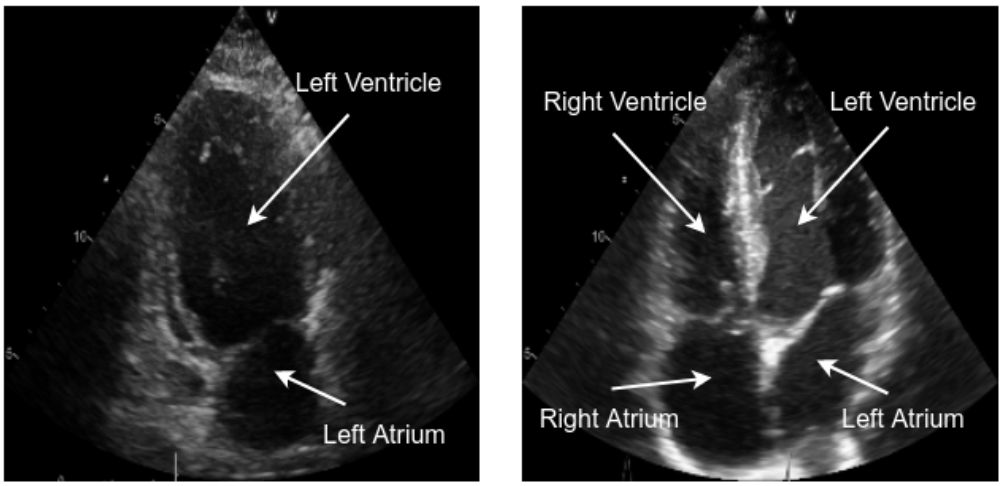
Figure 2.2: An ultrasonic transducer in operation.

2.1.2 Diagnostic Echocardiography

A variety of organs and soft tissues can be imaged with ultrasound, including the human heart. By itself, cardiac US imaging also has a large number of applications: it is used for general heart examinations, as well as before, during, and after surgical operations. Naturally, the diagnostic applications (focused on in this work) are the most common, though the perioperative uses are also increasing in popularity with the development of three-dimensional echocardiography.

Diagnostic cardiac ultrasound often employs two-dimensional imaging modalities, yet even then there is a choice between two methods: the non-invasive *transthoracic echocardiography* (TTE) and the invasive *transesophageal echocardiography* (TEE). TTE is regarded as the "normal" way of imaging through the chest wall, while TEE is performed by passing the transducer down the patient's esophagus. TEE is resorted to when a TTE examination is insufficient, either because certain regions are unreachable, or the images are too unclear to make a proper diagnosis. Other types of echocardiography involving small transducers passed through a catheter (intracardiac and intravascular) are now rising to prominence, but these are mostly perioperative procedures.

When performing TTE, one can look at the heart from different *cardiac views* (geometric perspectives). One view can provide information about the heart that is simply unattainable from other views. Over time, the most informative views have been identified and given their own names. Thus, some of the most widely used types of views are: parasternal, apical, subcostal and suprasternal. Nearly each of these view types is then sub-divided into additional categories. For example, there are two-, three-, four-, and five-chamber apical views (the five-chamber view images the four actual heart chambers and the left ventricular outflow tract). See Figure 2.3 for examples of the two- and the four-chamber views.



(a) Apical two-chamber view *(b) Apical four-chamber view*

Figure 2.3: Examples of the apical two-chamber view (A2C) and the apical four-chamber view (A4C). Depicted heart chambers are labeled.

2.2 The Left and the Right Heart in Cardiology

The human heart has four chambers, two of which are on the left side (left ventricle and left atrium), with the remaining two on the right side (right ventricle and right atrium). The left chambers receive oxygenated blood from the lungs and send it to the rest of the body, while the the right chambers receive deoxygenated blood and send it to the lungs for oxygenation (Figure 2.4).

One beat of the heart is divided into two phases: *diastole* and *systole*. During the diastole phase, the heart relaxes, and blood enters the ventricles. During the systole phase, the heart contracts, and blood leaves the ventricles. The heart is a double pump, meaning that relaxation and contraction occur simultaneously on both sides.

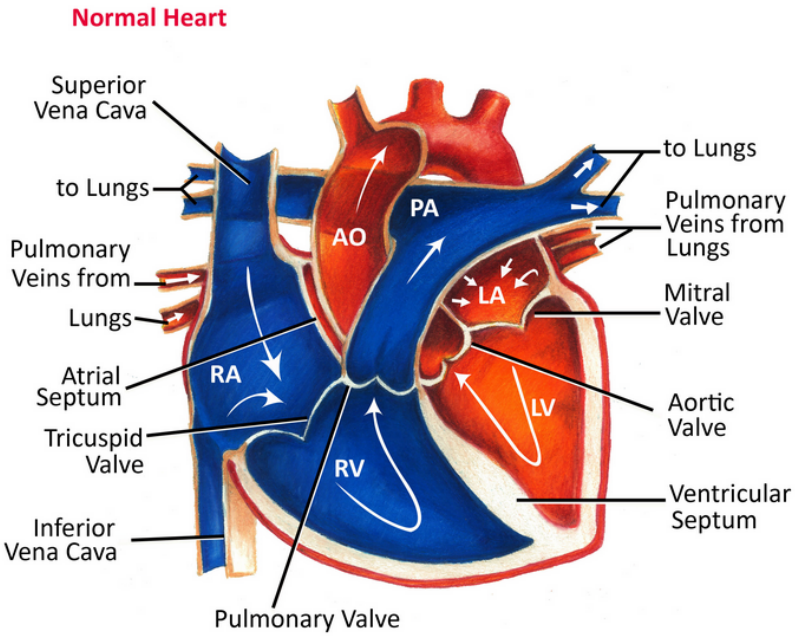


Figure 2.4: A simplified diagram of a normal heart [7]. Heart chambers and blood flow are depicted. In actuality, the left and the right sides are not completely symmetric.

2.2.1 Function and Role of the Left Heart

The left side of the heart is responsible for pumping oxygenated blood to body tissues and organs. Blood first enters the left atrium (LA) from the lungs via the pulmonary veins, then proceeds into the left ventricle (LV) through the mitral valve (diastole phase), and finally escapes through the aortic valve to supply the rest of the body (systole phase). Function of the left heart is often assessed via metrics associated with the left ventricle, meaning that this chamber is particularly interesting to cardiologists. Given the relationship of the left heart to all organ systems, it is no surprise that reduced function of the left ventricle can lead to a wide array of potentially fatal problems. For example, cardiac failure correlates with changes in the left ventricle [8].

Failure of the left ventricle can occur due to any number of causes, and establishing the exact reason requires careful analysis on the doc-

tor's part. However, measuring left ventricular function is less difficult. Some of the metrics that comprise such a measurement are the end-diastolic left ventricular volume (LV_{EDV}), the end-systolic volume (LV_{ESV}), and the ejection fraction (LV_{EF}).

Left ventricular ejection fraction refers to the percentage of blood that leaves the ventricle after each contraction. *Kosaraju et al.* explain that there are several classification systems of the left ventricular function by LV_{EF} values [9]. The simplest classification by the *American College of Cardiology* defines LV_{EF} between 50% and 70% as normal, with the values above 70% considered hyperdynamic, and the values below 50% indicating various states of dysfunction (the lower LV_{EF}, the more severe). One way to calculate LV_{EF} is as follows:

$$LV_{EF} = \frac{LV_{EDV} - LV_{ESV}}{LV_{EDV}} \tag{2.1}$$

When the heart's output is not sufficient to meet the body's needs, heart failure occurs. Cardiologists distinguish between two types of heart failure by LV_{EF} values: with reduced ejection fraction (HFrEF), and with preserved ejection fraction (HFpEF). The former type of failure points to abnormalities in heart contraction (systolic failure), while the latter indicates possible issues with heart relaxation (diastolic failure, but not always). Both types of heart failure seem to appear with nearly equal frequency [10]. Thus, measuring LV_{EF} alone can help diagnose HFrEF cases that constitute about half of all cardiac failures. As for the other half made up by HFpEF cases, LV_{EF} is ineffective at detecting them, and other metrics need to be used instead. Even then, measuring LV_{ESV} and LV_{EDV} on their own may allow cardiologists to judge whether LV hypertrophy has occurred - a possible sign of HFpEF [11].

Although the importance of the left ventricle can not be overstated, there are cases where paying attention to the left atrium may also be needed. Along with LV hypertrophy, LA enlargement can be a sign

of HFpEF specifically, whereas LA dysfunction of any kind may be a marker of various heart conditions in general. Furthermore, a very recent work by *Bisbal et al.* argues that atrial dysfunction is often neglected by researchers, and that "atrial failure" should be classed as a clinically relevant entity to foster better understanding of atrial dysfunction [12]. The authors state that LA plays a large role in LV filling and the total performance of the heart, meaning that "atrial failure" could be considered the primary cause of certain disorders.

2.2.2 Function and Role of The Right Heart

The right side of the heart receives deoxygenated blood from the body and sends it to the lungs. The entry occurs via the superior vena cava (blood from the upper body) and the inferior vena cava (blood from the lower body) into the right atrium (RA). In the diastole phase, blood flows from the RA into the right ventricle (RV) through the tricuspid valve. In the systole phase, blood escapes the RV into the pulmonary trunk through the pulmonary valve.

Compared to its counterpart in the left heart, the RV seemed to attract little attention from cardiologists and researchers, according to *Voelkel et al.* [13]. Dysfunction of this heart chamber was often considered a byproduct of disease, with the actual cause lying elsewhere (in the LV, for example). However, the RV plays an important role in pulmonary hypertension, wherein the blood pressure of lung arteries becomes too high. Examination of the RV can help diagnose this disorder. Unfortunately, familiarity with ultrasound techniques for imaging the RV also used to lag behind, and therefore echocardiographic examinations would sometimes omit the RV and the right heart entirely [14]. Since much of the research focusing on the RV acknowledges these issues, awareness has likely been increasing over the last decade. As for the RA abnormalities such as enlargement, they are usually associated with similar pulmonary diseases that also affect the RV as well as rare

congenital defects [15].

A case could be made that a much bigger focus on the left heart in cardiology and medical ultrasound is justifiable. If right-sided heart failure is indeed often caused by left-sided heart failure, then research on the left heart understandably takes precedence. The difficulty of imaging the right heart caused by its complex geometry and anterior position only reinforces the utilitarian logic in this regard. Lastly, pulmonary hypertension is somewhat rare, with the incidence rate of all its forms estimated to around 326 per 100,000 people by one study in Armadale, Australia [16]. Out of this number, 77% is the cases associated with left heart dysfunction. The incidence rate for all cardiovascular diseases is hard to estimate, but averaging the data from 34 European states collected by *Eurostat* in 2015-2018 puts it at around 2,000 per 100,000 inhabitants [17].

There are cases beyond pulmonary hypertension where the RV becomes dysfunctional, however, and it may be there that the importance of examining the right heart becomes evident. *Gorter et al.* argue that RV dysfunction is yet another important marker for HFpEF, bringing up several studies where a link between the two was observed [18]. They also maintain that RV dysfunction, when present along with HFpEF, contributes to poor prognosis and mortality risk. Therefore, treatments targeting the right heart could potentially prolong the life of some patients with HFpEF. The authors provide a variety of metrics to diagnose RV dysfunction by, among which is $RV_{EF} < 45\%$.

There may be a surge in interest towards the RV in light of the COVID-19 crisis. *Park et al.* review the role of RV in COVID-19, noting that RV dysfunction is a major predictor of mortality for patients with acute respiratory distress syndrome (ARDS) - a syndrome that SARS-CoV-2 infection may cause [19]. Pulmonary hypertension may also accompany ARDS caused by the virus. *Bleakly et al.* likewise indicate that critically ill COVID-19 patients often show signs of RV dysfunction when judged by the Fractional Area Change (FAC) metric [20].

FAC is similar to ejection fraction, but uses area instead of volume for calculations.

2.2.3 Heart as a Complex Singular System

Dividing the heart into two sides or four chambers is convenient for exploring the functions of each individual part, but it distracts from the fact that the heart is one organ. As previously suggested, the LV and the RV can affect each other and form a complex relationship. Normally, the RV assumes a semilunar shape and occupies a smaller volume than the elliptic LV. However, RV enlargement as a result of pulmonary hypertension is not isolated - it affects the LV as well (see Figure 2.5).

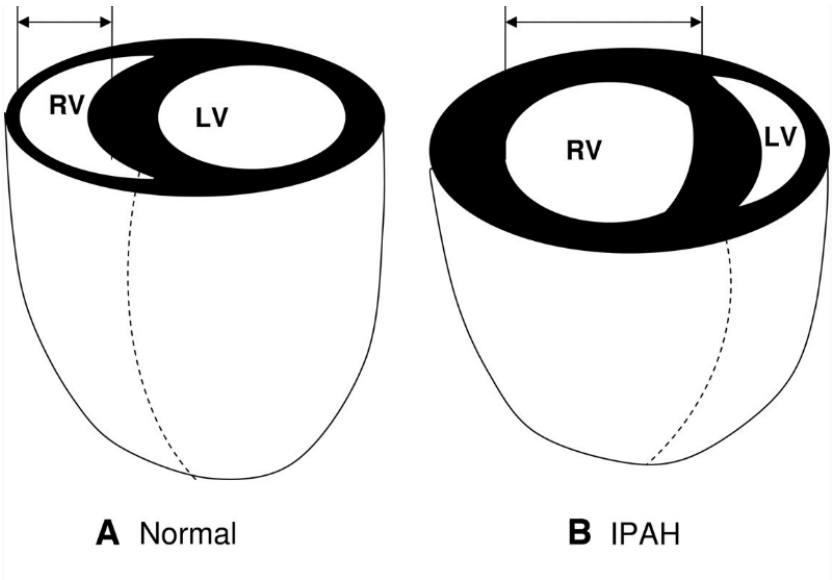


*Figure 2.5: The approximate shapes of LV and RV in a healthy heart (**A**) and when affected by idiopathic pulmonary arterial hypertension (**B**) [13].*

The interdependence of the LV and the RV is great enough for researchers to ponder whether LV failure and RV failure are truly separate phenomena. *Friedberg and Redington* mention that LV contraction is responsible for possibly more than half of the mechanical work

done by the RV, while the RV geometry influences LV function in turn [21]. The ventricles also share the septum on the inside and the myocardial fibers on the outside. These findings serve as an additional argument for the need to research the right heart and monitor its function during examinations.

2.3 Image Analysis with Artificial Intelligence

Artificial intelligence is an older field of research than most would think, with the first proof of concept being presented at the 1956 Dartmouth College Artificial Intelligence Conference [22]. Since then, the field has seen rises and falls of enthusiasm. Initially, the computing power limitations made complex intelligent algorithms unfeasible, leading to a decline in the amount of research. In recent decades, advancements in hardware sparked newfound interest in AI. Today, intelligent algorithms are used in many industries, from banking to medicine, and in people's daily lives as well.

Currently, the sheer variety of AI algorithms makes it difficult to even keep up with the new developments. There are methods for analyzing any kind of data: tables, text, sound, human speech, and images. One unifying quality for all of them is that they *mimic* biological systems and processes, at least in part. For example, intelligent methods aimed at image analysis are inspired by the workings of the visual cortex - part of the brain responsible for visual information processing.

2.3.1 Machine Learning and Artificial Neural Networks

Machine learning is an application of artificial intelligence that allows a computer to solve a certain task without having an explicitly pro-

grammed solution. Instead, the algorithm learns from its errors and refines its approach to the task over time. Machine learning can be further subdivided into *supervised* and *unsupervised* learning.

Supervised learning techniques require labeled data for the algorithm to learn from. In image analysis these techniques can be employed for object recognition, classification, segmentation, or regression (predicting measurements or values from an image). Unsupervised learning techniques, on the other hand, do not require labeling of the data. These techniques are normally used for clustering data according to similarities discovered by the algorithm. If a task can somehow be solved with either type of learning, supervised learning tends to produce better results given the appropriate setup. At the same time, collecting and labeling data is tedious work that often has to be delegated to humans. Ensuing subjective labeling and data set imbalance may introduce a bias to the model. Since this work focuses primarily on supervised learning techniques, some preliminary data analysis will be required to uncover possible limitations of the trained models.

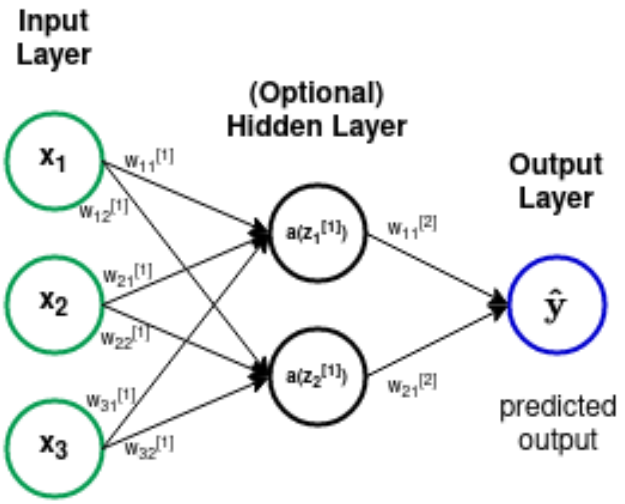


Figure 2.6: The basic structure of a three-layer Artificial Neural Network.

Artificial Neural Networks (ANNs) are an example of supervised

learning techniques. As seen from the name, ANNs take inspiration from neuroscience and use a connected network of so-called *artificial neurons* or *nodes* for calculations. Like their natural counterpart, ANNs are rather flexible and can solve a variety of tasks (depending on implementation specifics). The basic structure of an ANN (Figure 2.6) sees artificial neurons arranged into two or three layers:

- **Input layer:** data are fed into the network at this layer, with each *feature* or dimension of the data being treated as one node.
- **Optional hidden layer:** after multiplication with a *weight* matrix **w**, data arrive at this layer (if present), and an *activation function* a is applied. This layer serves to enhance the computational ability of the network, and the number of nodes is arbitrary.
- **Output layer:** the procedures occurring at the hidden layer are repeated at this layer. The number of nodes corresponds to the desired dimensionality of the output.

The process of propagating data from the input to the output layer is called *forward propagation*. Its mathematical description for each node is as follows:

$$x_j^{[l]} = a(z_j^{[l]}) = a\left(\sum_{j=1}^{N^{[l-1]}} w_{ij}^{[l]} x_i^{[l-1]} + b_i^{[l]}\right) \tag{2.2}$$

where $x_i^{[l]}$ is the value of the i-th node in the layer $l-1$, w_{ij} values are weights, b_i are biases (trainable offsets, not conventional biases), $N^{[l]}$ is the number of nodes in the layer, and $z_j^{[l]}$ is the output of the node. Then, the final output is obtained by applying the chosen activation function a to $z_j^{[l]}$.

A forward propagation step allows the network to produce a hypothesis, but the network has to be trained for its predictions to become more reliable. During training, forward propagation concludes by calculating the error through the chosen *cost function* (or *loss function*)

$C(\hat{\mathbf{y}}, \mathbf{y})$. The loss needs to be calculated in order to perform a *backpropagation* step, where the network learns from the errors by updating weights and biases:

$$w_{ij}^{[l]} \leftarrow w_{ij}^{[l]} - \eta \delta_j^{[l]} x_j^{[l]} \tag{2.3}$$

$$b_j^{[l]} \leftarrow b_j^{[l]} - \eta \delta_j^{[l]} \tag{2.4}$$

where η is the *learning rate*, δ_j are loss function gradients, and $x_j^{[l]}$ are inputs at layer l, i.e. activations of the previous layer $l-1$. The calculation procedure for δ_j in the last layer L is

$$\delta_j^{[L]} = a'(z_J^{[L]}) \frac{\partial C}{\partial (y_j^{[L]})} \tag{2.5}$$

whereas for all preceding layers l it is slightly more complex:

$$\delta_j^{[l]} = \sum_i \delta_i^{[l+1]} w_{ij}^{[l+1]} a'(z_j^{[l]}) \tag{2.6}$$

Note that the weight and bias update procedure are not set in stone, but are "pluggable", just as the activation and cost functions. Eq. (2.3-2.4) merely describe one of the simpler *optimizers*, stochastic gradient descent (SGD) optimizer.

The training process is essentially a cycle of forward and backpropagation steps on different samples of data. Ideally, the loss should decrease as the training continues, but the wrong choice of activation function, loss function, or optimizer can interfere with learning, as can any number of other factors. The relative complexity of the above equations may lead one to believe that mathematical mistakes are also a frequent occurrence, but contemporary software libraries tend to abstract away most of the difficulty on that front.

The flexibility of ANNs means that they *can* be applied to image data, but it is rarely a wise choice. Conventional ANNs reserve a "neuron" connection for every single feature of a data sample, and images have an extremely high dimensionality. Consider a relatively small

100×100 pixel image in RGB (three color channels) format: a computer sees it as a numeric array with 30,000 values, each of which would require a "neuron". Larger images can have millions of features, making the dense connectivity of ANNs computationally expensive.

2.3.2 Convolutional Neural Networks

Unlike ANNs, *Convolutional Neural Networks* (CNNs) assume that the input is an "image" - a numeric array with two or more dimensions. CNNs are more optimized for this type of information than ANN, requiring far fewer learning parameters to achieve better performance on image-related tasks. Instead of densely connected layers, these networks feature convolutional layers that apply "sliding" filters (or kernels) with learnable values to regions of an image. For a grayscale image with one color channel, a convolutional layer works like a 2D cross-correlation operation:

$$z[m,n] = \sum_{i=0}^{D-1} \sum_{j=0}^{D-1} w[i,j] * x[m+i, n+j] \tag{2.7}$$

where the $[i,j]$-like notation denotes a value in the i-th row, j-th column of an array; x is the input image, w is the filter ("weight") that the image is convolved with, and z is the layer output. Eq. (2.7) assumes that the filter has an equal height and width $D = 2n+1, n \in \mathbb{N}$. See Figure 2.7 for an illustrated example.

For images that have more than one color channel (RGB images, for example), the filter depth has to match the number of channels - the result is still a 2D feature map. A convolutional layer can also apply several trainable filters to an image, and the output depth will be equal to the number of filters applied.

Convolutional layers are frequently followed by *pooling* layers in CNN that also apply sliding kernels, but these kernels either aver-

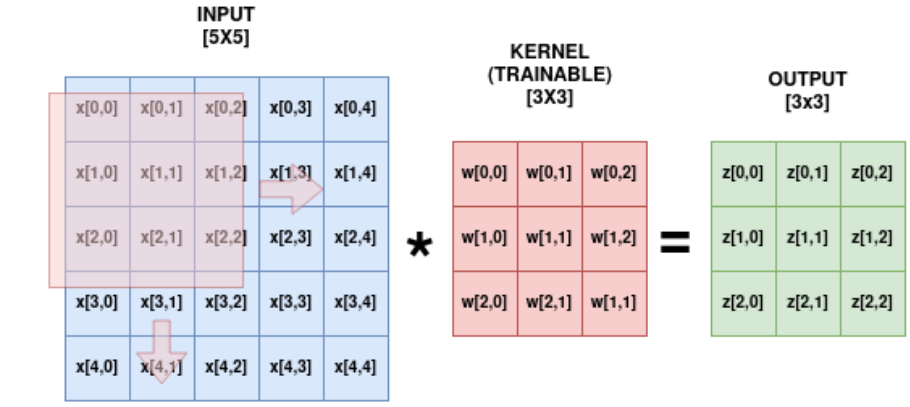


Figure 2.7: An illustrated example of convolving a 5×5 image with a 3×3 kernel.

age the values in the inspected region (average pooling) or select the maximum value (max pooling) instead of cross-correlating (Figure 2.8). Pooling layers serve to downsample their input to boost translational invariance and help recognize features no matter where they appear in the image.

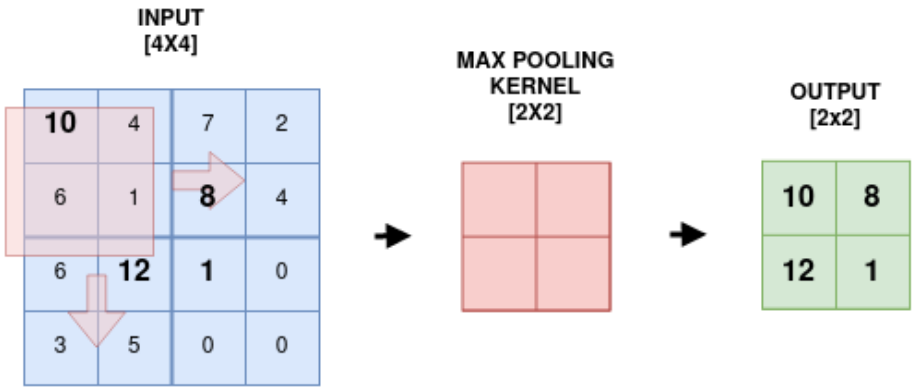


Figure 2.8: An illustrated example of max pooling a 4×4 image using a 2×2 kernel

It is important to note that both convolutional and pooling layers do not preserve the original height and width of the input. The dimensions of the result depend on the filter/kernel size, but can also be manipulated through *padding* and *striding*. Padding is often applied with convolutional layers to prevent the input size from changing, and increased striding (> 1) makes kernels skip parts of the input to reduce overlaps in the output.

Interestingly, the combination of convolutional and pooling layers appears to resemble the processes occurring in the visual cortex of animals, where similar functions are performed by biological cells of different types [23]. Convolutional and pooling layers can also be stacked several times over to produce various Deep Convolutional Neural Network (DCNN) architectures - a subject of research in *deep learning*. This practice came into popularity in 2012 with the introduction of AlexNet, a DCNN that stacks five "blocks" consisting of a convolutional layer, a max pooling layer, and a ReLU activation function [24]. In this kind of network, consecutive layers learn to recognize increasingly more complex features, starting with simple lines in the first layer and progressing to complicated shapes, such as human face outlines, by the final layer. Later research on human brains revealed that this mechanism of cascading calculations with increasing feature complexity at each stage also bears similarities to how we recognize objects [25].

For tasks like classification, DCNNs are usually constructed with one or more densely connected layers (featured in ANN) at the end. The previously mentioned AlexNet architecture uses this approach as well. In classification-oriented DCNNs, dense layers are applied in order to transform feature maps into scores or probabilities for each class. Other tasks, including segmentation, are better solved by DCNNs that do not include dense layers - such networks are also called Fully Convolutional Neural Networks (FCNNs).

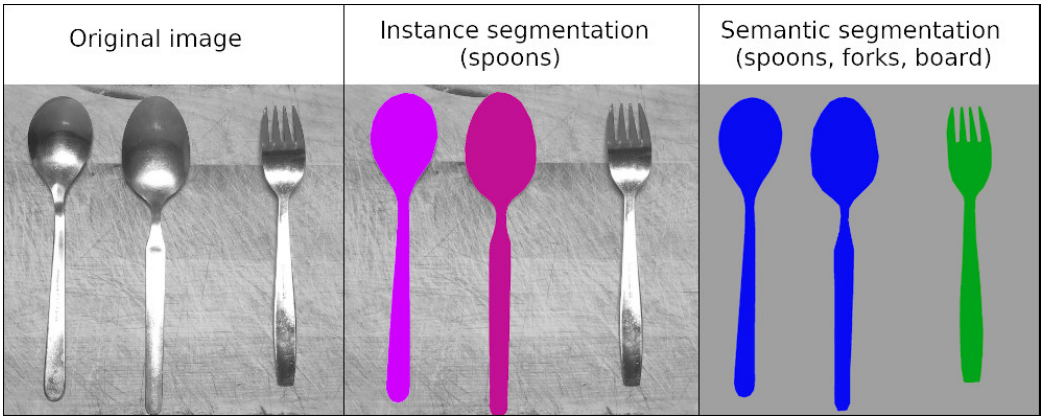


*Figure 2.9: Segmentation of an image with two spoons and a fork on a cutting board. Instance segmentation targeting spoons identifies two spoon instances (colored differently). Semantic segmentation targets object **types** instead (both spoons are therefore colored the same).*

2.3.3 Semantic Segmentation and Decoder-Encoder Networks

In image analysis, segmentation is a task of classifying each pixel in an image as belonging to an object *or* a type of object. This distinction gives rise to two kinds of segmentation tasks: instance segmentation and semantic segmentation. The former kind aims to find all instances of the same object type in the image and classify them distinctly, while the latter only distinguishes between object types and disregards individual instances (see Figure 2.9 for a visual demonstration).

This work focuses on semantic segmentation, as it seems to be the preferred approach in biomedical segmentation tasks. When considering echocardiograms specifically, semantic segmentation is also more appropriate than instance segmentation, because there are no objects of the same type that appear more than once.

Semantic segmentation can be performed with FCNNs, as mentioned previously, but there are certain caveats. Normally, it is desired that the segmentation result has the same size as the original image, meaning that layer parameters (convolution kernel size, padding, and

striding) should be carefully selected. Another problem is computational resources: CNNs *are* more efficient at image analysis than ANN, but the number of learning parameters can rise rapidly in deep networks.

One way to keep the number of learning parameters down while keeping up good segmentation performance is to arrange the layers into an *encoder-decoder* architecture. Encoder-decoder networks consist of an *encoder* part that downsamples the input (fewer learning parameters), and a *decoder* part that upsamples the result, restoring the original dimensionality.

This work relies on a particular encoder-decoder architecture called U-Net, as proposed by its inventors *Ronneberger et al.* [26]. U-Net is intended for segmentation of biomedical images, and it works well with echocardiograms. The original architecture of U-Net features four encoder layers that form a contraction path, a bottom layer serving as the bottleneck, and four decoder layers as the expansion path (for a total of five "levels"). Each encoder layer also has a skip connection to a corresponding decoder layer, resulting in a structure with a "U"-like shape (Figure 2.10).

The standard U-Net architecture is modifiable to a great degree. For example, contraction and expansion layers can be added or removed, the initial number of feature maps can be increased or decreased, upsampling can be performed via a standard interpolation algorithm or configured as additional trainable layers performing a transposed convolution operations.

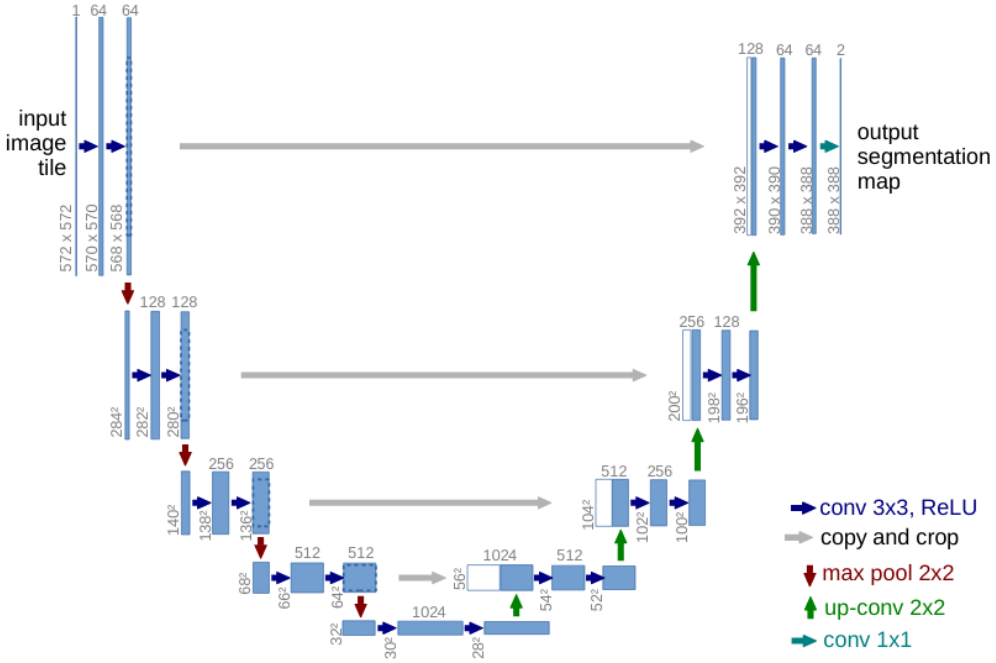


Figure 2.10: The standard U-Net architecture by Ronneberger et al. [26]

2.3.4 Transfer Learning

Transfer learning is a family of techniques that allow the knowledge accumulated by AI algorithms in a *source task* to be applied to a different (but still related) *target task*. The main purpose of transfer learning is to improve generalization and overall performance of the algorithm in solving the target task. Additionally, the algorithm may improve faster with transferred knowledge and/or require less data relevant to the target task.

A good example of transfer learning for image classification or object recognition is fine-tuning a model that has been already pre-trained on a large, extensive data set such as ImageNet [27]. ImageNet is an extremely large data set commonly used for benchmarking, as it contains over 14 million images spread over 1,000 classes. Pre-training on ImageNet (or a sufficiently large subset of it) tends to produce models that

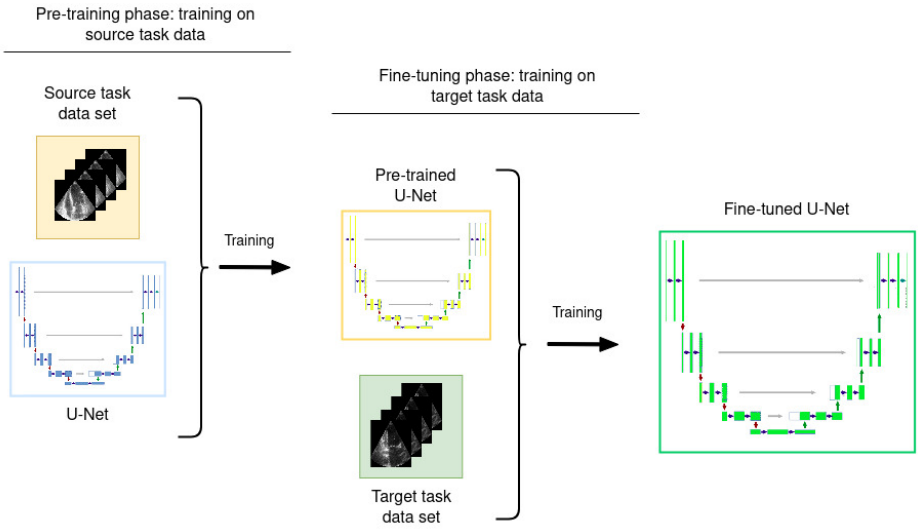


Figure 2.11: An example of a simple transfer learning scheme with U-Net

recognize a lot of general features, which makes them useful for a wide range of tasks. These models can then be reused to identify only a few of the original classes, or to recognize entirely new classes.

In general, transfer learning with (F)CNN is mostly understood as a kind of weight initialization procedure, with certain additions. The bare-bones process is as follows: a neural network is first trained for the source task, then its state is saved, the data set is switched to that of the target task, and the network continues training (Figure 2.11). This scheme will be used during the transfer learning experiments later in this work.

Although more complex procedures are outside the scope of this thesis, it is possible to modify the process. For example, certain weights could remain fixed after pre-training, so that only a part of the network continues training in the fine-tuning phase. In the example with ImageNet pre-training, all layers but the last one (the classifier) could be "frozen" to enable complete reuse of all learned features. However, it is not immediately obvious which layers should be frozen, if any. For this reason, even more complex algorithms like adaptive fine-tuning

are devised, allowing parts of a neural network to be frozen depending on the input [28].

2.4 Artificial Intelligence in 2D Echocardiography

AI offers solutions to a great number of applications within diagnostic 2D echocardiography and possesses the potential to automate many of the involved sub-routines, if not entire heart examinations. The existing AI models can (to an extent) predict which cardiac view an echocardiogram belongs to, segment heart chambers in certain views, estimate chamber volumes and other clinical metrics, and detect diseases. Performing these computations is possible in real time as well, meaning that intelligent support can be provided during examinations. AI may also be capable of adhering to industry and measurement standards more precisely than humans, thus reducing the amount of inter-observer variability [2, 3].

Quite a few of AI models or systems have been proposed for the purposes of echocardiography at the time of writing. There are two striking examples: the multi-task data pipeline designed by *Zhang et al.*, and the more focused system of *Smistad et al.* that derives certain LV measurements in real time [29, 30].

The automated echocardiogram interpretation system of *Zhang et al.* appears to be a truly monumental undertaking. It consists of multiple deep learning models trained for specific subtasks that pass their predictions to each other (Figure 2.12). The videos of echocardiographic examinations are first sent to a model that attempts to classify the view used in the video (among 23 defined views), then the video frames are segmented if the video belongs to one out of the five supported views. The system uses a separately trained segmentation model for each view type. Segmentation results are used to derive LV size, mass,

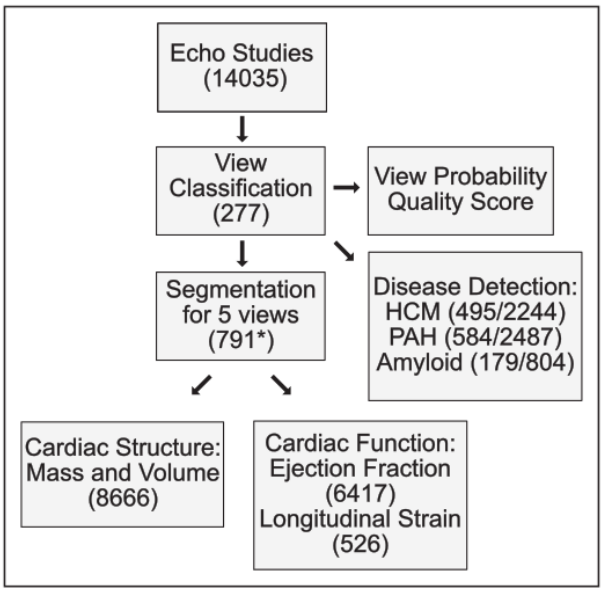


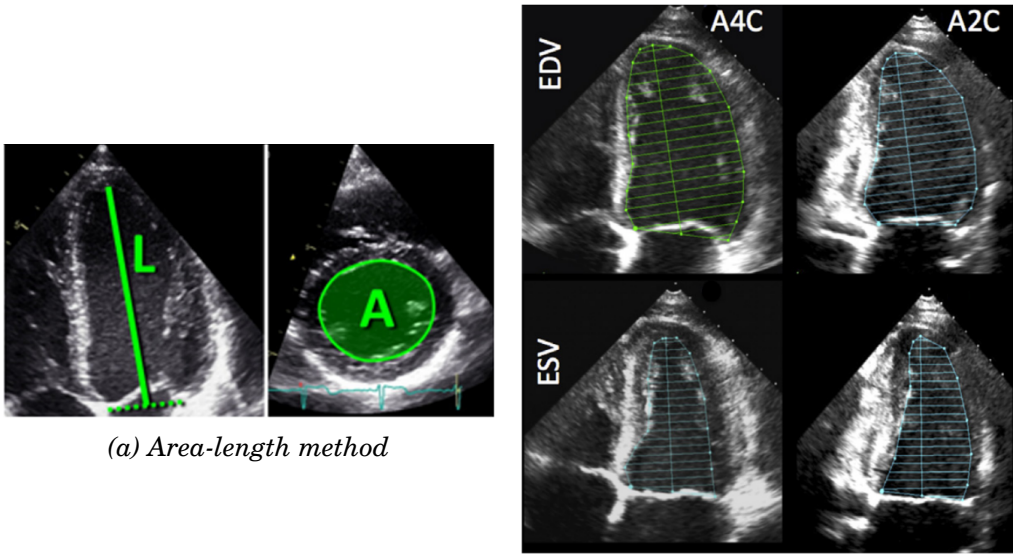
Figure 2.12: The description of the data pipeline by Zhang et al. The number in brackets is the number of echocardiograms used for model training [29].

and volume - and calculate LV_{EF} along with other metrics in turn. Alternatively, videos can be fed to models that identify certain diseases such as hypertrophic cardiomyopathy (HCM), pulmonary arterial hypertension (PAH), and cardiac amyloidosis.

While the system devised by *Smistad et al.* does not have the same number of capabilities, it does its work in real time, processing up to 43 frames per second. It can distinguish between seven different view types, and allows automated segmentation of two: A2C and A4C views. Similarly to the other system, segmentation results provide the basis for deriving LV_{EF}.

Both systems rely on AI for image segmentation, but LV volumes and LV_{EF} are derived from the images *without* using AI. Instead, the calculations are performed using recommended methods for such tasks, with *Zhang et al.* choosing the area-length method and *Smistad et al.* opting for the biplane disc summation method (Figure 2.13) [31]. Once the LV volumes at end-diastole (LV_{EDV}) and end-systole (LV_{ESV}) are

derived, either system can estimate LV_{EF} through Eq. (2.1).



(a) Area-length method

(b) Biplane disc summation method

Figure 2.13: The LV volume calculation methods approved by the American Society of Echocardiography [31]. The biplane disc summation method needs A2C and A4C views. It is preferable to the area-length method, because it traces the LV shape instead of assuming the circumference area is constant.

In contrast to the works *Zhang et al.* and *Smistad et al.*, there are systems that use AI to directly predict LV volumes and even LV_{EF}. For instance, *Ghorbani et al.* propose a data pipeline that identifies certain "regions of interest" in an echocardiogram [32]. These regions are highlighted by AI and may indicate LV hypertrophy or LA dilation. They may also reveal the presence of a catheter, a pacemaker, or a defibrillator leads. The system can estimate LV_{ESV} and LV_{EDV} in order to calculate LV_{EF}, but it works even better when predicting LV_{EF} straight from the image.

Another example is the work by Ouyang et al. that uses a complex but elegant system for LV_{EF} estimation and HFrEF prediction [33]. The system analyzes entire videos, performing LV segmentation and estimating LV_{EF} with a three-dimensional (spatio-temporal) CNN. The

variation in segmented area over time is then used to identify cardiac cycles, allowing more accurate LV_{EF} values to be obtained over several heart beats. Data samples with estimated $LV_{EF} < 50\%$ are marked as expressing cardiomyopathy.

When it comes to research on the right side of the heart, there seem to be only a few available works. One relevant and recent study by *Karuzas et al.* details an experiment on automated RV segmentation with good results, yet only the abstract of the article is published at the time of writing [34]. When it comes to automated quantification of RV function, the study of *Beecy et al.* appears successful in using similar deep learning methods to examples above, but the authors note that there is very little research in this area [35].

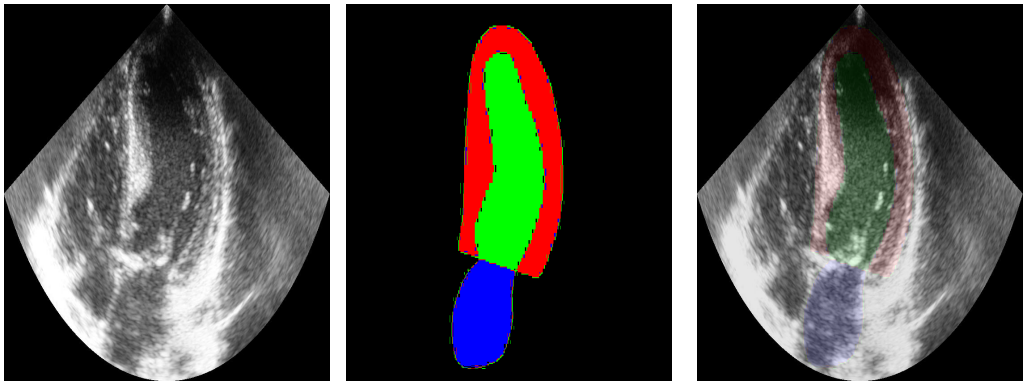
Chapter 3

Overview of Acquired Data

This chapter introduces the 2D echocardiogram data sets for use in further experiments. Three data sets in total are explored: two of them focus on the left heart structures, while the last one contains segmentation masks for the right side of the heart. Section 3.1 covers the properties of CAMUS data set - an extensive, publicly available data set of 2D echocardiograms with labels for LV endocardium, LV epicardium, and LA, as well as additional information about the patients (the information is anonymized). Section 3.2 introduces a similar but smaller data set provided by GE Healthcare, code named as GE data set. Then, Section 3.3 discusses a data set (code named RV data set) with segmentation masks for RV endocardium, RV epicardium, and RA, also compiled by GE Healthcare for this work.

3.1 CAMUS Data Set

At the time of writing, the most extensive publicly available data set of 2D echocardiograms appears to be the Cardiac Acquisitions for Multi-structure Ultrasound Segmentation (CAMUS) data set [36]. Interestingly, the authors themselves raise the concern that there is barely any competition on this front. Of the works on AI-assisted echocar-



(a) Original image *(b) Ground truth mask* *(c) Overlay*

Figure 3.1: A sample from the CAMUS data set. From left to right: end-diastole image in A4C view ***(a)****; ground truth mask showing LV epicardium (red), LV endocardium (green), and LA (blue)* ***(b)****; mask overlay* ***(c)****.*

diography surveyed in the previous section, only that of *Ouyang et al.* discloses the training data (EchoNet-Dynamic database). However, the video resolution of that data set is quite low (112 $\times$ 112 px), and only the LV epicardium is annotated.

The CAMUS data distinguishes itself with a varying but high image resolution. It includes cardiac cycle sequences from 500 patients of the University Hospital of Saint-Etienne, France. The sequences are both the two-chamber and the four-chamber apical views. Unfortunately, the complete information is provided only for the training subset (450 patients). The images that correspond to the end-diastole and the end-systole parts of the cycle (1,800 images) were annotated by an expert who manually segmented the regions occupied by the left atrium (LA) as well as the epicardium (LV_{epi}) and the endocardium (LV_{endo}) of the left ventricle (see Figure 3.1). The remaining 15,464 mid-sequence images were not annotated. The data set includes the clinical metrics for every patient (LV_{ESV}, LV_{EDV}, and LV_{EF}).

3.1.1 Additional Properties of the CAMUS Data

In addition to clinical metrics, the CAMUS data set contains information about patients' age and sex. Assessing this information may reveal underlying bias in the algorithms trained on the CAMUS images. The insights on patients' age (see Figure 3.2) indicate that the data are not suited for any use in pediatrics (data was taken from adults only). Young adults are also under-represented, with only 9 out 450 patients (2%) being below the age of 30. Inspecting the biological sex of the patients shows that 292 patients (65%) are male, while only 158 (35%) are female. Unfortunately, information about patients' height and weight is not provided in the data set.

According to a study by *Pfaffenberger et al.*, factors like sex, age, height, and weight have some correlation with heart size [37]. Among the four factors, biological sex has an especially strong influence, as it accounts for almost a 9.5ml difference in LV end-diastolic volume. Additionally, men's hearts have a somewhat lower normal LV_{EF} values (52-72%) than women's (54-74%) [9]. As a result, the hearts imaged in the CAMUS data may be somewhat larger on average than they would be if male/female representation were more balanced. Lower LV_{EF} values would be expected too, but the creators of the data set took care to balance the representation of normal and abnormal value ranges.

3.1.2 CAMUS Data Preparation

The images in the CAMUS data set are not uniform in quality: the authors divide the sequences into those of "Poor", "Medium", and "Good" quality. This metric has no relation to image resolution (see Figure 3.3), but instead represents the clinical accuracy of the images - whether they were taken from a "good" view or not. Annotations and LV_{ESV}/LV_{EDV}/LV_{EF} calculations are subsequently affected as well.

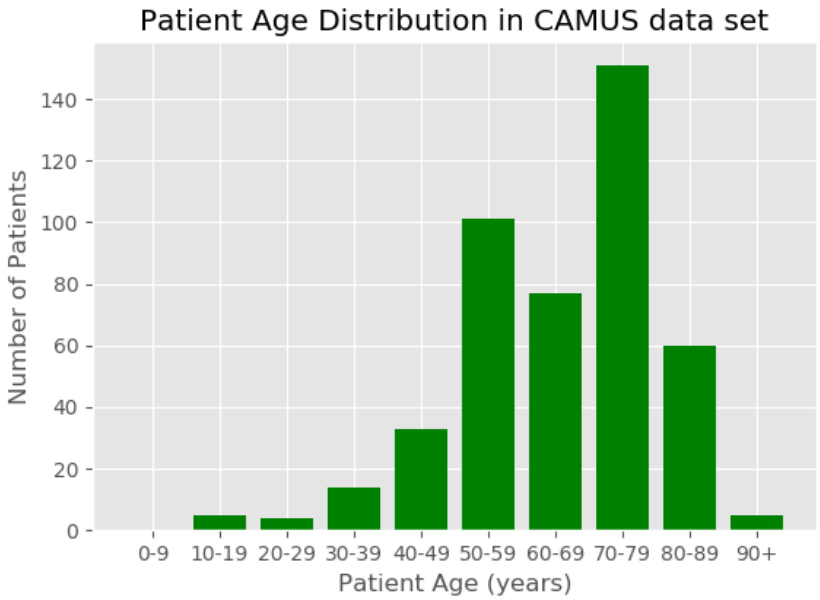


Figure 3.2: The patient age distribution in the CAMUS data set

Leclerc et al. claim that images of "Poor" quality (from 59 out of 450 patients) are not useful for clinical purposes and remain in the data set only to study their effect on the training process of AI models. When analyzing the experiment results, the authors conclude that these images neither contribute nor derive from the model performance to any significant degree. Removing this part of the data set reduces it from 1,800 to 1,564 annotated images in total.

As already demonstrated, CAMUS images have a lot of variability in terms of resolution. Such data cannot be used "as is": its dimensionality must be standardized. To that end, there are at least three approaches to choose from:

- pad all images to match the highest resolution (within a batch or across the whole data set);
- split data into patches of equal resolution;
- simply resize all data to a standard resolution.

The approach of padding would preserve the dimensionality of rele-

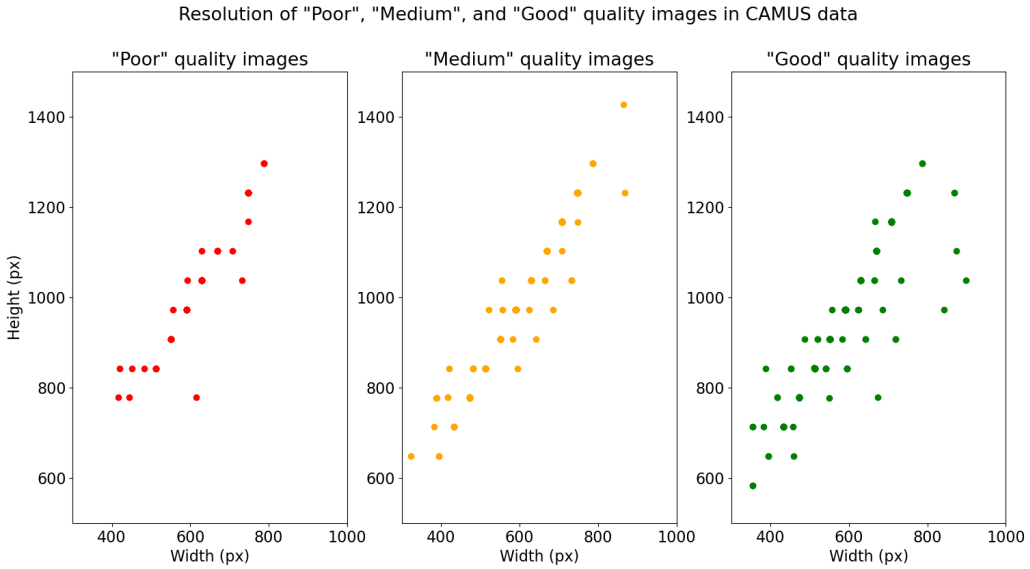


Figure 3.3: The image dimensions in the CAMUS data for each patient. From left to right: "Poor", "Medium", and "Good" quality images. Images of any quality are found across the whole graph, meaning that image quality and image resolution are indeed unrelated. Some points are superimposed due to identical resolution. The "Good" quality graph (on the right) excludes ***one*** *extreme outlier at (1181, 1945).*

vant data, but introduce a lot of junk information (zero-pixels) and slow down the computation. Splitting the data into patches would reduce the dimensionality per data sample, yet much of the spatial information would be lost. Finally, resizing the data would be near-effortless, but it is sure to introduce minor artifacts and distortions (especially in the ground truth masks) due to imperfect interpolation. However, such distortions should only become an issue in case of *overfitting*, wherein an AI model starts to perfectly match the training data instead of looking for the underlying patterns. Given the sheer complexity of echocardiogram data, the risk of overfitting seems rather small, especially if preventive policies are introduces during training. Therefore, resizing the whole data set is likely the best choice. The final argument in favor of this approach is that the other data sets used in this work (introduced later) were all provided at a standard image resolution of 256 $\times$ 256 pixels.

3.2 GE Left Heart Data Set

Using only the CAMUS data set for the left heart data would make it unclear whether the findings in this work are generally applicable. Therefore, the thesis will rely on one more left heart data set provided by GE Healthcare (unavailable to the public). This data set appears to be sourced from Padua University Hospital, Italy (336 images) as well as Rigshospitalet, Denmark (229 images). The subsets were annotated by cardiologists from the respective institutions - one expert per data subset. The annotation format is nearly identical to that of the CAMUS data, featuring LV_{epi}, LV_{endo}, and LA contours. Still, there are a few crucial differences between the properties of CAMUS data set and this one:

- The GE data is a mixed set of A2C and A4C cardiac views, while CAMUS data contains both views for each patient.
- The GE data includes annotated images at both end-diastole

and end-systole, but not for every patient - some images are not "paired".

- The GE data has a standard resolution of 256 $\times$ 256 pixels, while CAMUS data is not uniform in this regard.
- The GE data does *not* contain clinical metrics such as LV_{ED}, LV_{ES}, and LV_{EF}. Information about patients is not included either.

With the above considerations in mind, it becomes clear that this data set cannot be analyzed the same way as CAMUS, because it contains far less information. It is difficult to make judgements about the underlying bias in the data without knowing anything about the patients whose hearts were imaged.

The labels (segmentation masks) of this data set are different from CAMUS. Whereas CAMUS masks maintain color intensities of 0 (background), 1 (LV endocardium), 2 (LV epicardium), and 3 (LA), GE masks use intensities of 0, 85, 170, and 255 for the same classes (Figure 3.4). Dividing the color values by 85 brings the masks to same format as CAMUS, though the procedure may not be strictly necessary, depending on how the input to the neural network is standardized.

Because there is no guarantee that every end-diastole image has an end-systole pair and vice versa, it may be prudent to remove the unpaired patient data, and then to merge the subsets from both institutions. As a result of these operations, 540 images from 270 patients remain in total.

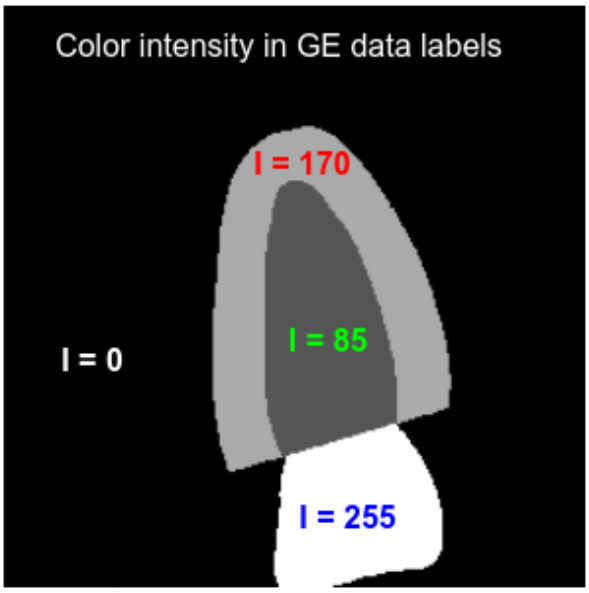


Figure 3.4: Color intensities in the original labels of the GE data set.

3.3 RV Data Set

The final data set is a mix of RV-focused images and four-chamber view images labeled by GE Healthcare specifically for use in this work. The data set will be referred to as the **RV data set** for brevity and convenience, though labels for RA are also present. After checking for presence of an end-diastole image where an end-systole image is available for the same patient and vice versa, 446 paired images with labels remain. As with the GE data set, all images are provided at 256×256 pixel resolution.

The labels for each class (RV_{endo}, RV_{epi}, and RA) are stored as separate images, however, meaning that they had to be unified to produce masks that are similar to CAMUS and GE data. These labels also lacked a sharp cutoff, making the boundary ambiguous. A decision has been made to include all pixels with intensity above 0 (on the scale of 0-255) as belonging to the label, but it may have unforeseen consequences. Figure 3.5 provides an overview of the label unification

procedure, while Figure 3.6 offers an example of an RV-focused image and a four-chamber view image with integrated labels.

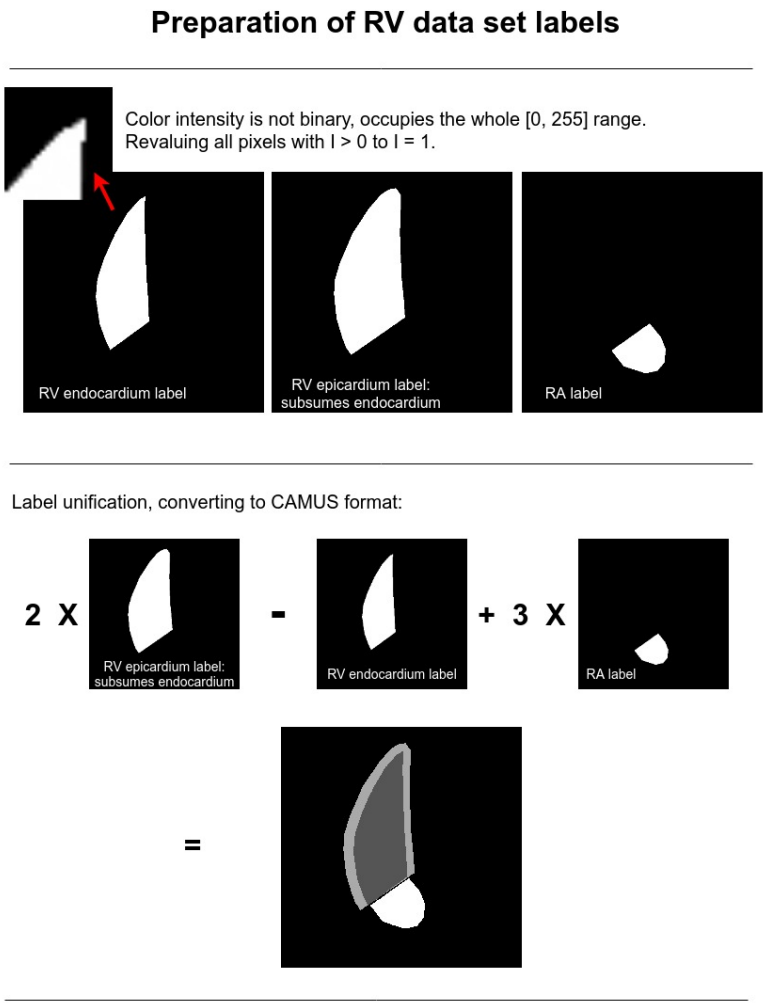
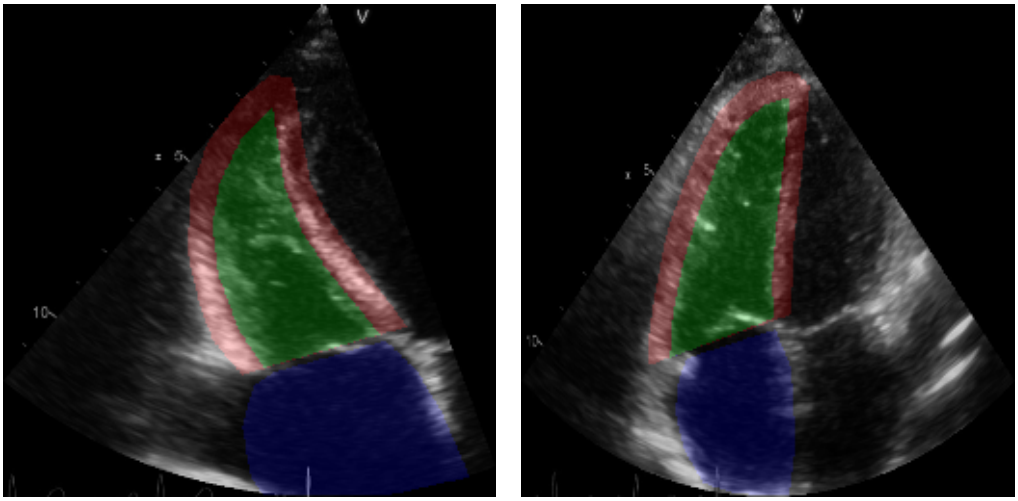


Figure 3.5: Label preparation in the RV data set. All pixels with above zero color intensity are included in the label. The labels for different heart regions are unified into complete masks (the same format as the CAMUS and the GE data).



(a) An RV-focused image *(b) A four-chamber view image*

Figure 3.6: An example of an RV-focused image and a four-chamber image from the RV data set. Ground truth masks are blended with the original images (80% original, 20% mask).

Some of the images in the RV data set are sourced from Rigshospitalet, Denmark - the same clinical site that contributed to the GE data set. The remaining images are taken from a data set of athletes, but further details are unknown.

Chapter 4

Methodology

This chapter introduces the reader to the specific methods, algorithms, and parameters that will be used for echocardiogram segmentation, performance measurement, and knowledge transfer in the next chapter. The reasoning behind the choices is also explained, where appropriate. Section 4.1 introduces the custom implementation of U-Net used in later experiments and discusses its properties. Section 4.2 covers the use of Dice Coefficient both as a segmentation performance metric and a loss function for U-Net training. Section 4.3 demonstrates how a trained U-Net model can be applied to 2D echocardiograms to produce a human-readable segmentation. Finally, Section 4.4 discusses parameters and variables that can affect transfer learning and offers an experimental setup for measuring its effectiveness.

4.1 U-Net for Echocardiogram Segmentation

Segmentation experiments in the next chapter rely on a reproduced U-Net architecture (see Section 2.3.3) with some departures from the original architecture (Figure 4.1). For example, padding is introduced

to convolutional layers to preserve height and width of the input after the operation. Thus, the two dimensions are only affected during downsampling (with max pooling) and upsampling (with transposed convolution operations). As a result, the data dimensionality is more predictable between the layers, and the output mask has the same height and width as the input image. Furthermore, *batch normalization* is added after each convolution as a regulator to improve the network's accuracy and ability to generalize [38]. Batch normalization performs *z-score normalization of the data*:

$$z(\mathbf{X}) = \frac{\mathbf{X} - \mu(\mathbf{X})}{\sigma(\mathbf{X})} \tag{4.1}$$

where $\mathbf{X}$ is the input data sample, $\mu(\mathbf{X})$ is the mean of the data, and $\sigma(\mathbf{X})$ is the standard deviation. The result, $z(\mathbf{X})$ essentially replaces all values in the sample with the number of standard deviations they are away from the mean. Given the presence of batch normalization layers in the network, it is reasonable to regularize the input images in a similar manner.

The custom U-Net architecture supports a variable number of initial feature maps. Changing this hyperparameter drastically affects the total number of trainable parameters in the network (see Table 4.1).

U-Net Instance	Initial Feature Maps	Trainable Parameters	GPU Memory (per image)
U-Net 8	8	490,000	100 MB
U-Net 16	16	1,940,000	210 MB
U-Net 32	32	7,760,000	430 MB

Table 4.1: Overview of U-Net instances for experiments. Assumes 256×256 pixel inputs. The total number of trainable parameters and GPU memory consumption per image processed increase with the number of initial feature maps.

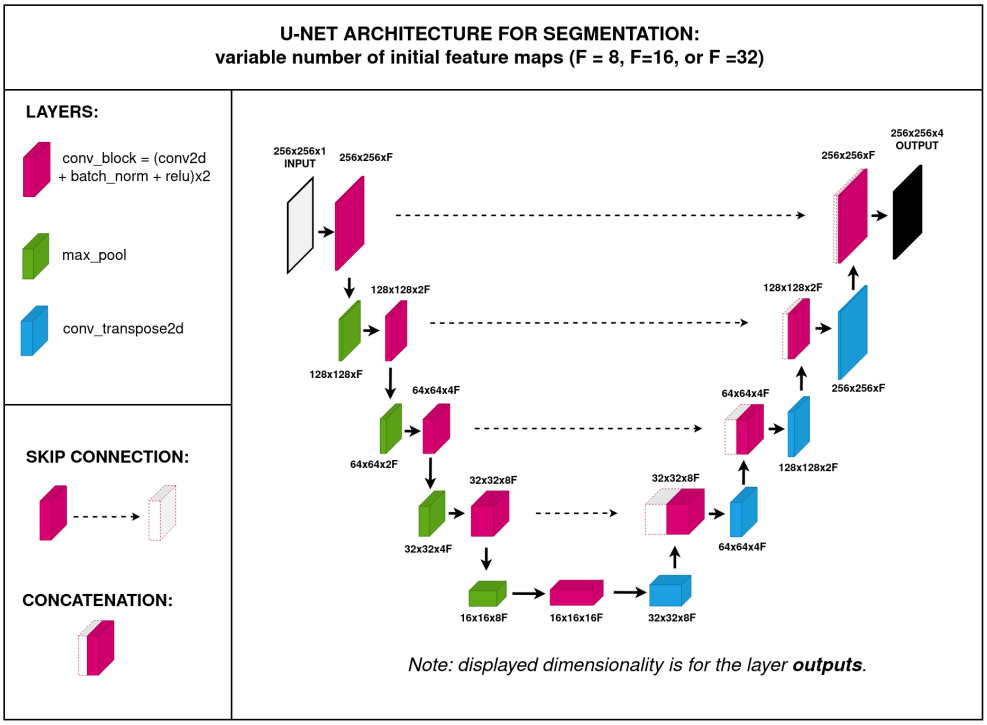


Figure 4.1: The reproduced U-Net architecture for experiments. Convolution and max pooling kernel sizes are the same as the origin. Padding is used to preserve width and height after convolutions. Includes batch normalization layers to improve generalization.

A higher number of trainable parameters should, in theory, lead to more precise segmentation models. On the other hand, GPU (or CPU) memory consumption would also increase, limiting the *batch size* - the number of images the network can learn from simultaneously. Limited batch size may result in erratic improvements during training and result in reduced performance. The original U-Net architecture of *Ronneberger et al.* features a rather large input size and the number of feature maps. Sticking to the same choices would likely raise the memory consumption to over 1GB per image and make it computationally prohibitive for all but the high-end GPUs currently on the market. When a model itself is saved as a file, the amount of memory it occupies is also affected. "Lighter" models lend themselves better to reuse, as they can be distributed more easily.

One additional detail to note is that batch normalization layers of the proposed custom U-Net heavily contribute to the total memory consumption. The original U-Net architecture probably did not make use of batch normalization, since the concept had only been introduced two months before the U-Net publication. Removing these layers would indeed make the network less computationally demanding, but the overall performance would suffer as well.

4.2 Measuring Segmentation Performance with Dice Coefficient

Adequately rating the performance of a segmentation model requires different metrics than the ones used for typical classification tasks. It is especially evident in echocardiogram segmentation, where most pixels may belong to the "nothing" (background) class - a severe class imbalance. Simply calculating the accuracy at pixel level would then provide misleading results: if every pixel is predicted as background, accuracy may remain still high (Figure 4.2).

To avoid such scenarios, the Jaccard Similarity Index or the Dice Similarity Coefficient may be used instead, along with a slew of other performance metrics [39]. The Jaccard Index and the Dice Coefficient are mathematically similar and positively correlated, meaning that using both is somewhat redundant. This work will mostly rely on the Dice Coefficient:

$$D = \frac{2|\mathbf{S} \cap \hat{\mathbf{S}}|}{|\mathbf{S}| + |\hat{\mathbf{S}}|} = \frac{2 * TP}{2 * TP + FP + FN} \tag{4.2}$$

where S is the ground truth mask and $\hat{\mathrm{S}}$ is the predicted mask. TP refers to True Positives - pixels that belong to a certain class and are predicted as such. FP denotes False Positives - pixels that are not in that class, yet still marked as belonging to it. Finally, FN are False Negatives - pixels that belong in the class, but are not predicted as

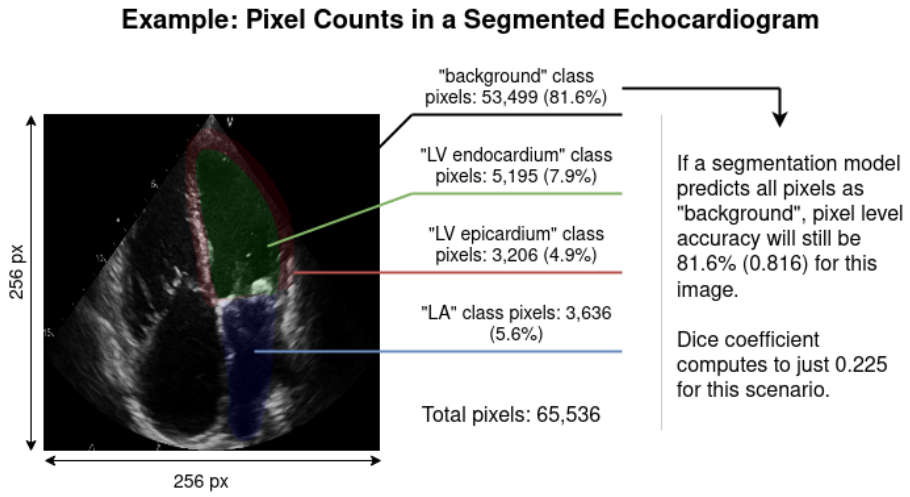


Figure 4.2: The disadvantage of using pixel accuracy as a segmentation performance metric. Example uses a 256×256 pixel echo image from GE data set. The Dice Coefficient is calculated as detailed in eq. (4.3).

such.

In a multi-class setting, both the true and the predicted masks can be treated as colored images - 3D arrays. At each color channel, matrices $\mathbf{S}_c$ and $\hat{\mathbf{S}}_\mathbf{c}$ would offer the ground truth and the prediction for the respective class c. Then, eq. (4.2) may be reinterpreted as follows:

$$D = \sum_c \left(\frac{2\left(\mathbf{S}_c \odot \hat{\mathbf{S}}_\mathbf{c}\right)}{\mathbf{S}_c + \hat{\mathbf{S}}_\mathbf{c}} \right) \tag{4.3}$$

where "$\odot$" is the element-wise multiplication operation. Note that the elements in $\hat{\mathbf{S}}$ are logits produced by *softmax* activation at the U-Net output layer. Since each element $\hat{s}_{c_{ij}} \in [0, 1]$, the overall score is more precise and continuous than with binary predictions. It is also possible to obtain a score for a particular data class alone by not performing the final summation in eq. (4.3):

$$D_c = \frac{2\left(\mathbf{S}_c \odot \hat{\mathbf{S}}_\mathbf{c}\right)}{\mathbf{S}_c + \hat{\mathbf{S}}_\mathbf{c}} \tag{4.4}$$

Lastly, the Dice Coefficient can be trivially refashioned into a loss function:

$$D_{loss} = 1 - D = 1 - \sum_{c} \left(\frac{2\left(\mathbf{S}_c \bigodot \hat{\mathbf{S}}_\mathbf{c}\right)}{\mathbf{S}_c + \hat{\mathbf{S}}_\mathbf{c}} \right) \tag{4.5}$$

Just like the Dice Coefficient, the Dice Loss can be viewed or reported for every individual data class.

4.3 Applying Trained U-Net Models to Individual Images

After the U-Net models are trained, they have to be useful for segmenting individual echocardiograms. The training process itself is already convoluted, but making the trained models play along and construct human-readable predictions requires some work as well. At the very least, all input has to follow the same format as during training, meaning that the images must be resized to 256×256 pixels and "z-normalized" by converting the pixel values to z-scores as detailed in eq. (4.1).

The output produced by the U-Net models is a four-dimensional array of logits (output of the softmax function in the final layer) with the following dimensionality: $(N \times C \times H \times W)$. The fact that there are four classes (background, endocardium, epicardium, and atrium), and the output values are logits, means that

$$\sum_{c=0}^{3} Y_{[n,c,h,w]} = 1.0 \tag{4.6}$$

where n refers to a specific image, and N - to a total number of images in the batch; c is the specific data class, and C is the total number of classes (four); h is the height value in pixels, and H is the total image height (256 pixels); w is the width value (in pixels), and W is the total

image width (also 256 pixels). Essentially, every value in the model output describes the "probability" or certainty that the model has about a particular pixel belonging to a given data class. This format does not lend itself very well for visualization. However, if the input consists of only one image, the number of dimensions is effectively reduced to three.

The output can be further processed with the *argmax* function. The function selects the highest value along a given dimension in the array and produces the index of that value. If the model receives only one image as input, and its output is fed through the argmax function, the result is a two-dimensional mask $(H \times W)$ dictating the most likely class that each pixel belongs to. Expanding the mask back into three dimensions and painting pixels with colors corresponding to each data class (green for endocardium, red for epicardium, and blue for atrium) would then produce a mask similar to Figure 3.1b. Finally, blending the mask with the original image gives a prediction that is easily understandable by humans. See Figure 4.3 for a summary of all the procedures. Applying trained U-Net models in this way and reviewing individual predictions helps to understand how the models make their decisions a bit better than only reviewing general statistics from the training process.

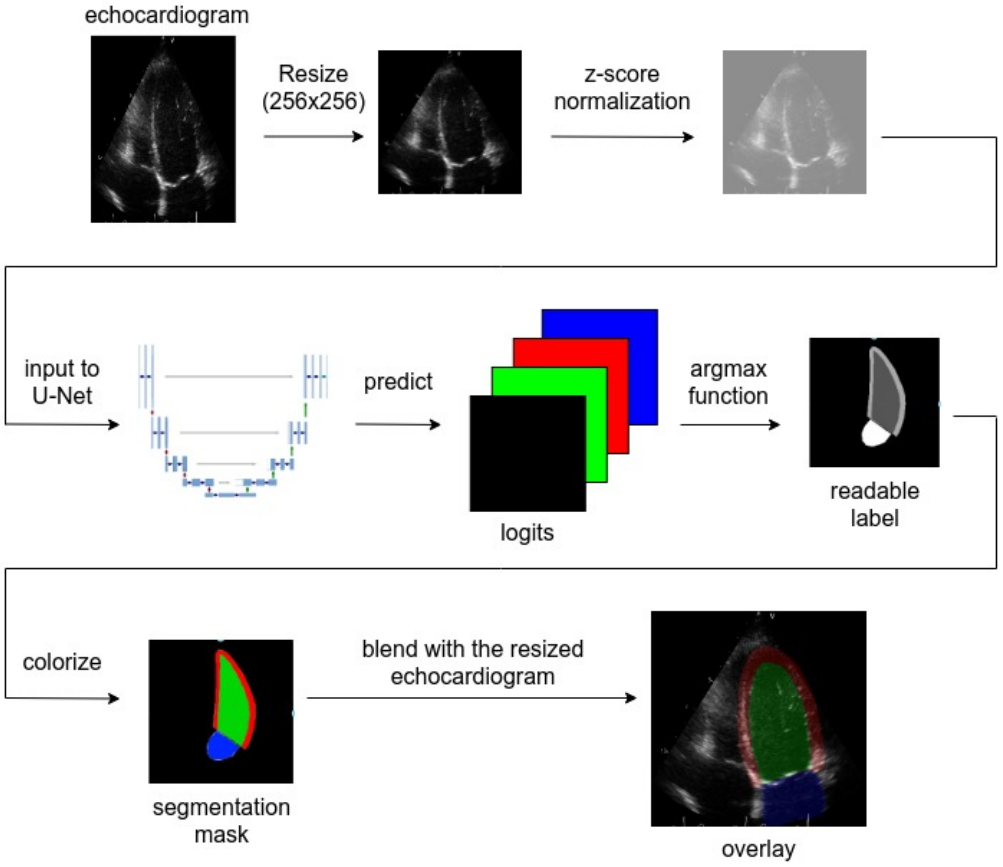


Figure 4.3: The summary of procedures for using trained U-Net models to segment echocardiograms.

4.4 Performance-affecting Variables, Assumptions and Benchmarking

Segmentation performance of the U-Net and the subsequent results of transfer learning are bound to be affected by a countless number of hyperparameters. How well any given U-Net instance does would at least depend on the number of encoder-decoder steps, number of initial feature maps, batch size, whether a simple interpolation or transposed convolutions are used for upsampling, how long the network trained, etc. For transfer learning performance, the list is even greater: the size of the target task data set plays a role, and so does the actual scheme of the knowledge transfer. Exploring the impact of all these variables is unrealistic with the amount of resources on hand, and the unavoidably complex visualization of the results would only confuse the reader. Certain "ground rules" are needed for the experiments: ones that take into account the resource limitations and the conditions where transfer learning is likely to be useful.

Transfer learning is a reasonable approach when a better AI performance on a task is desired, but only a small data set for that task is available, along with a larger data set for a "similar" task (see Section 2.3.4). Therefore, how little of the target task data is present makes for a parameter worth exploring. To maximize the potential positive impact of transfer learning, it makes sense to attempt pre-training (training on the source task) with a well-performing model. Furthermore, the complexity of the AI model is often a significant contributor to its overall performance. In the U-Net implementation for this work, the number of initial feature maps serves as a proxy for model complexity - it could serve as a good independent variable as well. Finally, when it comes to the duration of pre-training and fine-tuning, they should continue until the point where any further improvement is unlikely, so that every AI model achieves its full potential.

However, once fine-tuning is finished, how does one decide whether transfer learning is beneficial? Clearly, performance of fine-tuned models must be compared to that of models trained purely on target task data. Still, the "transferred" models get two opportunities to converge: in pre-training and in fine-tuning. In order to make the comparison fair, the "benchmark" models should be given an opportunity to train longer. One way to quantify the length of the training process is to consider the number of *epochs* - passes over the whole data set. Then, if a model is given N epochs to converge in pre-training and N epochs in fine-tuning, comparing it to a model that had $2N$ epochs to converge in training on the target task data seems reasonable. Resorting to this kind of epoch calculus is by no means ideal, but an ideal stopping criterion for training does not exist to begin with. A learning model can show improvement after very long periods of stagnating, but there is no way of predicting when or whether it will happen.

Given all the above considerations, a tentative set of parameters is proposed for the experiments (Table 4.2).

Experiment Parameter	Value(s)
U-Net Initial Feature Maps	8, 16, 32
Amount of data in fine-tuning and benchmark model training (images)	50, 100, 200, 400
Pre-training and fine-tuning duration (epochs)	100
Benchmark training duration (epochs)	200
Batch size (images)	10
Learning rate	1e-4
Optimizer	Adam
Upsampling scheme	Conv. transpose
Loss function	Multi-class Dice

Table 4.2: The proposed parameters for the transfer learning experiments

Chapter 5

Experiments and Results

This chapter is dedicated to conducting echocardiogram segmentation and transfer learning experiments. The procedures are first attempted with the two left heart data sets (CAMUS and GE) in Section 5.1 as a trial run of the algorithms and the experimental setup. Section 5.2 covers the main experiments - U-Net transfers between the left heart and the right heart data (CAMUS to RV as well as GE to RV). Finally, Section 5.3 looks at the predictions of individual images by the trained models to discern additional patters in their behavior.

5.1 Transfer Learning between Left Heart Data Sets

Before attempting transfer learning from left heart data to right heart data, it is prudent to confirm the validity of selected methods and parameters. Theoretically, transfer learning is easier to perform when the source task and the target task are very similar or even exactly the same. For this reason, knowledge transfer between CAMUS and GE data makes for a good trial run - both data sets are suited for LV and LA segmentation.

Section 4.4 discusses how the effectiveness of transfer learning can be analyzed. In this case, various U-Net models should be pre-trained on CAMUS data, then "benchmark" models are to be trained on GE data. Finally, the pre-trained models should be fine-tuned on GE data as well, after which their performance can be compared to the benchmark models.

5.1.1 Pre-training Models on CAMUS Data Set

U-Net models with different numbers of initial feature maps (F = $8, 16, 32$) were pre-trained for segmentation of CAMUS data. Of all data, 75% (1,173 images) were used for training, whereas the remaining 25% (391 images) were set aside for testing. For each configuration of initial feature maps (three configurations) ten trials were attempted, leading to 30 U-Net models trained in total. The models were given 100 epochs to train, but only the best performance in the testing phase was recorded and saved (testing phase occurs at the end of each epoch). Figure 5.1 displays the Dice Coefficient values for models with each configuration, averaged across ten trials.

As seen in the figure, the number of initial feature maps barely played any role in determining the model performance. It is likely that the duration of 100 epochs was more than enough for every U-Net model to converge to the best result possible for CAMUS data under the constraints of the experiment. The minuscule standard deviation values only appear to further reinforce this notion. While segmentation quality for LV epicardium and LV endocardium appears to lag behind the values for LA, all numbers are still rather high. To place the results in a context, one may review them alongside the work of *Leclerc et al.*, the creators of CAMUS data set [36]. Relevant values are displayed in Table 5.1.

However, *Leclerc et al.* only focus on the segmentation metrics for LV epicardium and LV endocardium, while also evaluating ED (end-

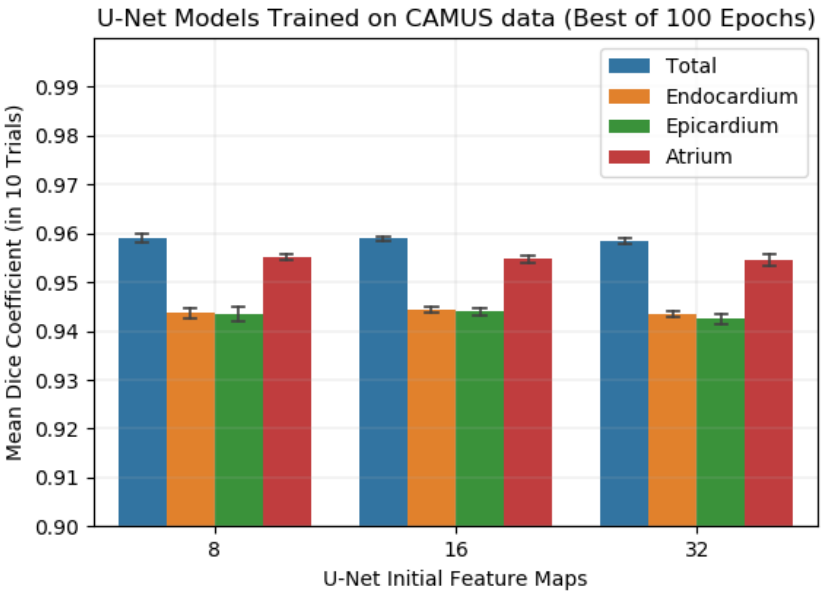


Figure 5.1: Mean Dice Coefficient values for U-Net models pre-trained on CAMUS data. Total multi-class Dice Coefficient values (including background) are shown along with specific values for each relevant data class: LV_{epi}, LV_{endo}, and LA. Standard deviation in the results is also shown in each case.

diastole) and ES (end-systole) images separately. Due to the differences in performance evaluation procedures, it is impossible to make a direct comparison. Thus, one cannot say for certain whether the U-Net models from this work performed better or not. Still, outperforming the previous works on segmentation of CAMUS data is not the goal of this thesis, and the achieved results are more than sufficient to proceed.

U-Net Configuration	Dice Total	Dice LV_epi	Dice LV_endo	Dice LA
U-Net 8	0.959 ± 0.001	0.944 ± 0.002	0.944 ± 0.001	0.956 ± 0.001
U-Net 16	0.959 ± <0.001	0.944 ± 0.001	0.944 ± 0.001	0.955 ± 0.001
U-Net 32	0.958 ± 0.001	0.943 ± 0.001	0.944 ± 0.001	0.955 ± 0.001
Configurations by *Leclerc et al.* [36]:				
U-Net 1 (ED*)	-	0.951 ± 0.024	0.934 ± 0.042	-
U-Net 1 (ES*)	-	0.943 ± 0.035	0.905 ± 0.063	-
U-Net 2 (ED*)	-	0.954 ± 0.023	0.939 ± 0.043	-
U-Net 2 (ES*)	-	0.945 ± 0.039	0.916 ± 0.061	-

Table 5.1: The Dice Coefficient values of U-Net models trained on CAMUS data. The results achieved by the creators of CAMUS data set are included for reference. Note that Leclerc et al. used a different, more detailed evaluation scheme, meaning that the values are not directly comparable. The numbers after "±" sign are the standard deviation values for U-Net 8, 16, 32; their meaning for U-Net 1 and 2 may or may not be the same. ***ES and ED refer to end-diastole and end-systole images being evaluated separately.***

5.1.2 Training Benchmark Models on GE Data Set

For benchmarking purposes, U-Net models were trained from scratch on the left heart data set provided by GE. Once more, three different configurations were used (U-Net 8, U-Net 16, U-Net 32). This time, however, one additional parameter was relevant: the amount of data in the training subset. The models were separately trained on 50, 100, 200, and 400 images respectively, with 100 images being reserved for testing in each case. Again, each combination of the parameters was evaluated across ten trials, leading to the total of 120 models trained. Furthermore, since both pre-training and fine-tuning are to provide 100 epochs for models to converge, these benchmark models were given 200 epochs of training instead for a fair comparison. The results are displayed in Figure 5.2.

Unlike with training on CAMUS, the amount of data used for this experiment is far more limited. It was therefore not surprising to see a greater variability in the results, which was especially prominent when training with merely 50 images. Interestingly, less complex models (U-Net 8) showed both worse performance and larger variability on average as well - it was not the case with CAMUS data set. On the other hand, the Dice Coefficient values were still unexpectedly high for all models. The last peculiarity was the lagging performance on LV epicardium segmentation combined with superb LV endocardium values. This behavior is potentially due to two reasons:

- In the images, the epicardium seems to occupy fewer pixels than the endocardium or the atrium on average. Therefore, the Dice Coefficient metric might punish the epicardium prediction errors more "harshly" than for other classes.
- In some images, the epicardium boundary stretches beyond the boundary of the regions detected by the ultrasound probe (Figure 5.3). These parts of the labels could be annotated arbitrarily, making them difficult to predict exactly.

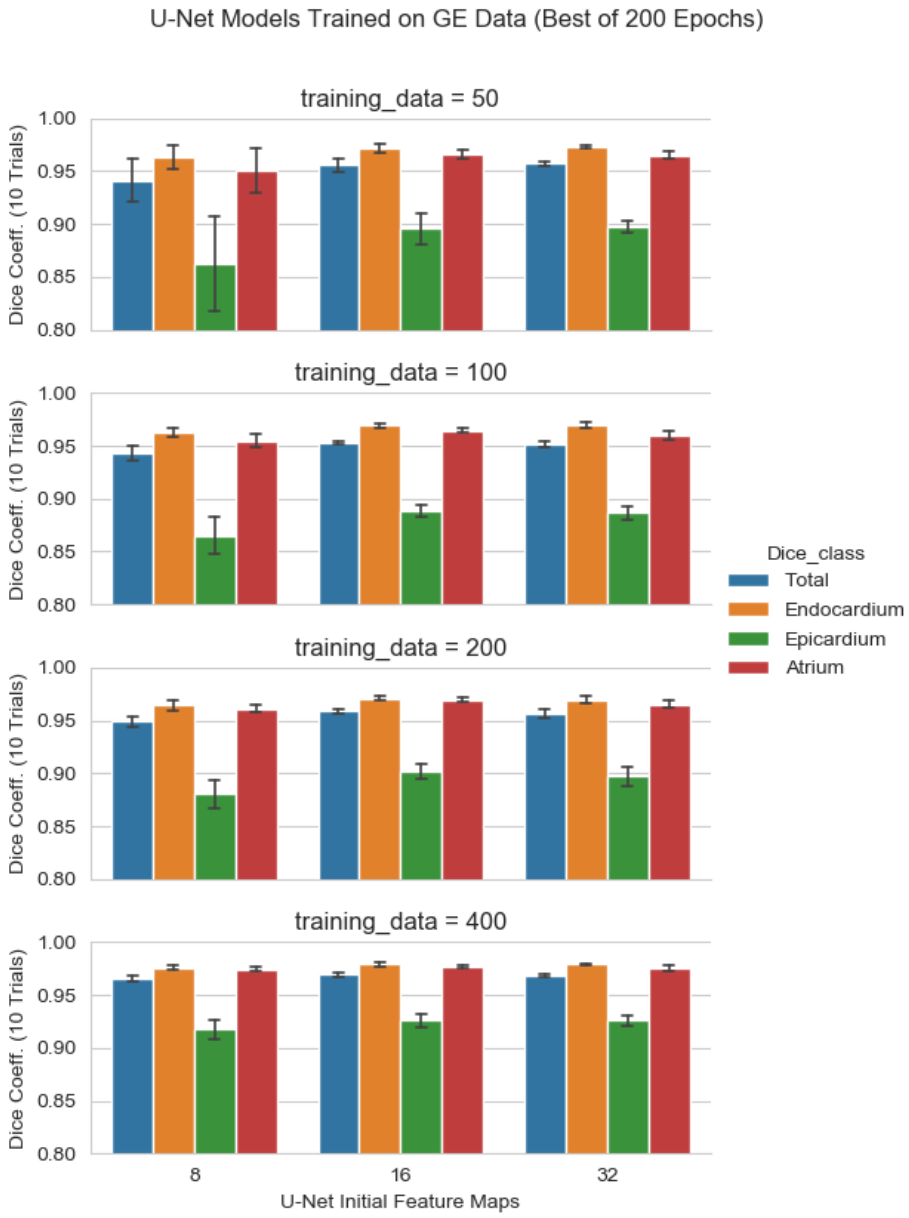


Figure 5.2: The Dice Coefficient values for benchmark U-Net models trained on GE left heart data. The results are grouped by the number of initial feature maps and amount of training data.

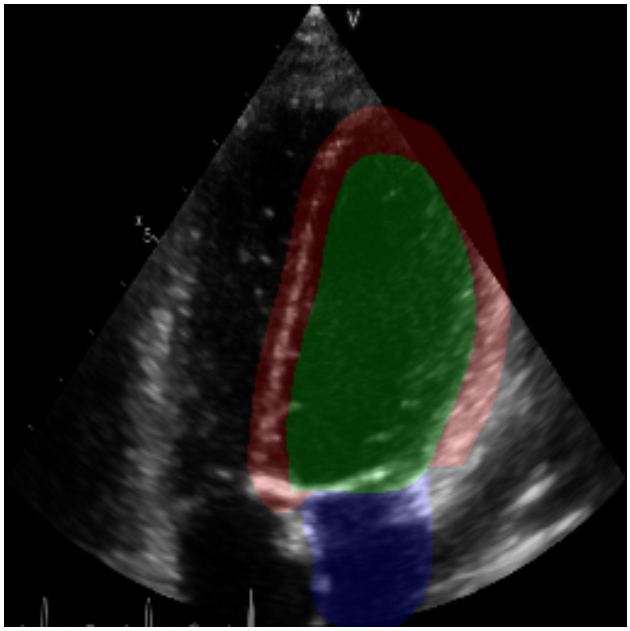


Figure 5.3: An example of the epicardium boundary stretching beyond the boundary of the "view" in the GE data set.

5.1.3 Fine-tuning Models on GE Data Set

The U-Net models pre-trained on the CAMUS data were fine-tuned by further training on the GE data. As with the benchmark models, both the number of initial feature maps and the amount of training data varied for this experiment. Essentially, each of the 30 pre-trained models continued training with 50, 100, 200, and 400 images of the GE left heart data set, thereby producing 120 new models. Once more, the size of the test data subset was kept to the same 100 images as with the benchmark model training. The results are shown in Figure 5.4.

Similarly to the benchmark models, the fine-tuned models had also struggled with segmentation of LV epicardium, but the overall score was markedly higher. The Dice Coefficient values actually increased for all data classes, reaching or even exceeding 0.990 in specific cases (mostly for LV endocardium). The variability in the results also decreased dramatically. The most striking detail is that all models

reached similar levels of performance, even the ones that could only train on 50 images. However, these models actually achieved slightly higher Dice score than the models that had access to more data, casting some doubt on the results. This kind of behavior would suggest that the training subset of 50 images might have favored the testing subset more than the larger training subsets. Similar hints can, in fact, be observed when comparing U-Net 16 and U-Net 32 benchmarks across different data set sizes in Figure 5.2.

Figure 5.5 provides a simplified overview of the performance differences between the fine-tuned models and the benchmark models with the same parameters. As expected, the increase in performance was negatively proportional to the amount of training data available for the target task. LV epicardium, being the most problematic area, also received the largest boost across all parameter combinations.

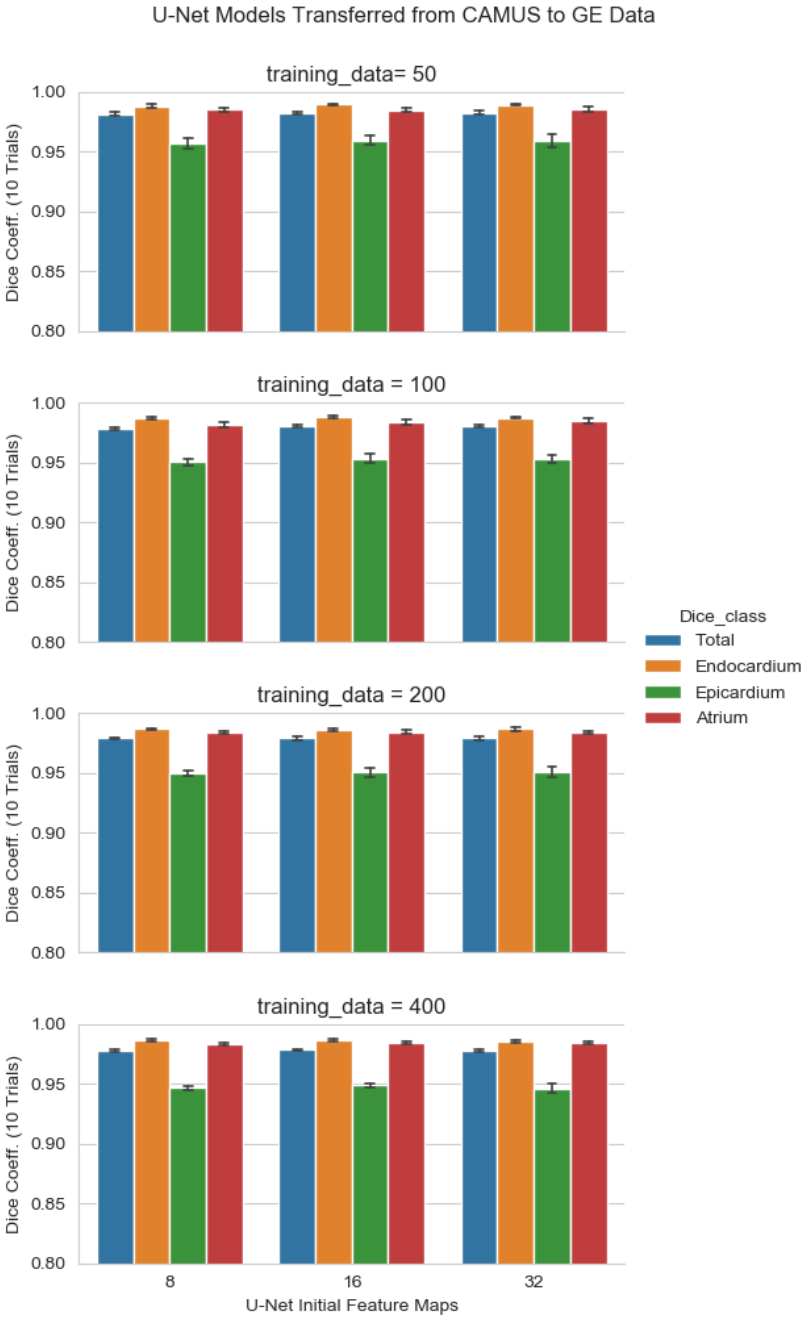


Figure 5.4: The Dice Coefficient values for U-Net models pre-trained on CAMUS data and then fine-tuned by further training on GE left heart data. The results are grouped by the number of initial feature maps and amount of training data.

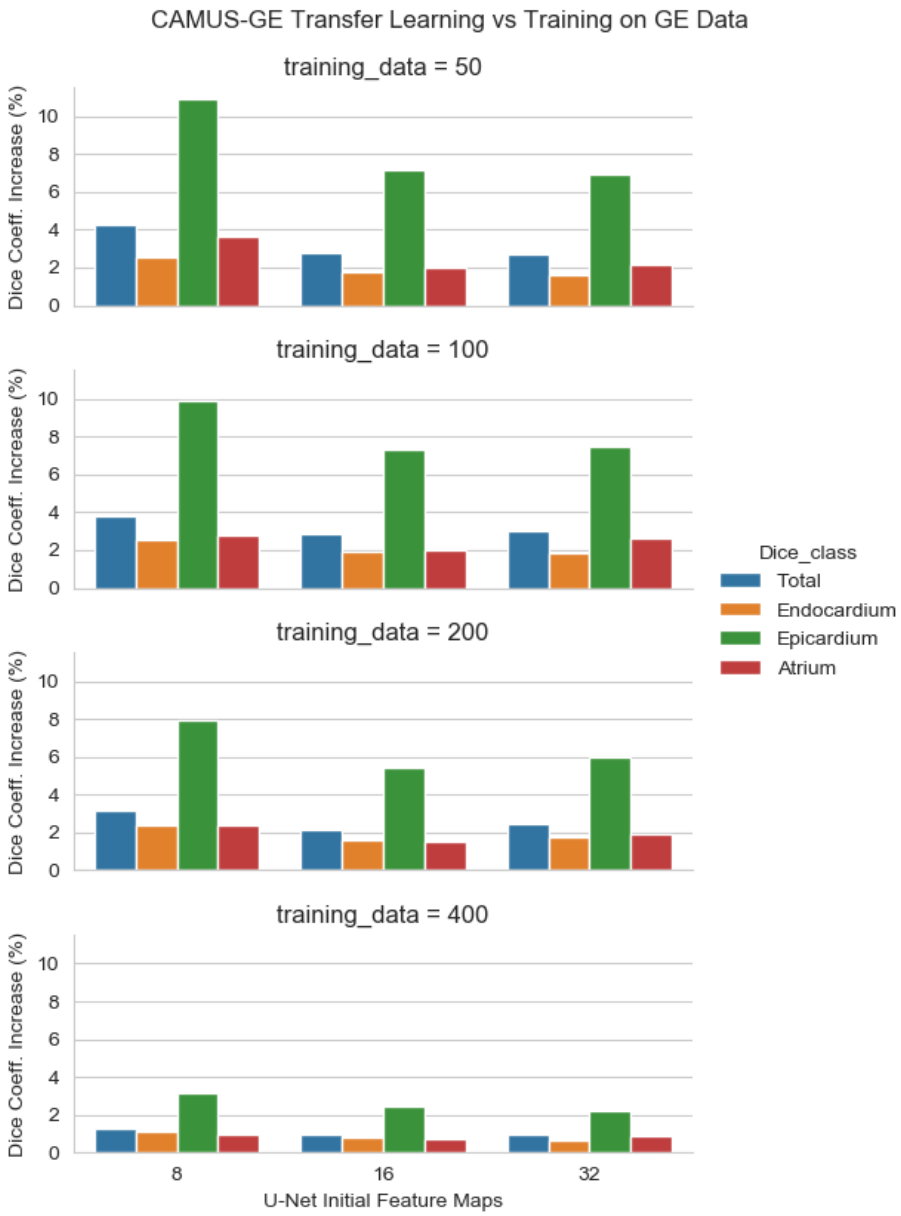


Figure 5.5: Comparison between Figures 5.2 and 5.4: the difference in the Dice Coefficient values (in percentage points) between the fine-tuned models and the benchmark models.

5.2 Transfer Learning from Left Heart to Right Heart Data

Investigating the feasibility of transfer learning from segmentation data for the left side of the heart to similar data for the right side of the heart is the main purpose of this work. This section details the experiments towards that goal. In practice, it means that U-Net models are first pre-trained on CAMUS or GE data sets and then fine-tuned on RV data set. However, some changes in the experimental setup are necessary, given the results from the previous section.

The transfer learning experiments between CAMUS and GE data served their purpose as a trial run. As seen in the Figure 5.5, the results produced by U-Net 8 are quite different from those of U-Net 16 and U-Net 32. Still, there is barely any difference between U-Net 16 and U-Net 32, which makes it clear that further experiments with U-Net 32 are superfluous. Excluding U-Net 32 models saves a lot of time and effort, as they are the most computationally demanding by far. Furthermore, the experiments with 400 training images cannot be replicated with RV data set, as there are only 446 images available, with 100 images reserved for testing (the data set was processed only after the conclusion of experiments in the previous section). Reducing the size of the largest training subset to 300 images seems to be reasonable under the given circumstances. All other parameters remain the same.

The pre-trained models may struggle with four-chamber images in RV data set, since their source task was to segment the left side of the heart in such images. For the fine-tuning to be successful, the models would have to "forget" this task and learn to focus on the right side instead. The outcome is not so obvious in advance, however, as some of the features learned by the models in the source task could potentially be reused.

5.2.1 Training Benchmark Models on RV Data Set

With pre-training on the CAMUS data having already been completed in previous experiments, there was no need to repeat the procedure. On the other hand, creating new benchmark models was still necessary, and 80 new U-Net models were trained on RV data set: U-Net 8 and U-Net 16 models, with 50, 100, 200, and 300 images in the training subset, 10 trials for each parameter combination. Once more, the total duration of training was 200 epochs, and the best performance was selected.

However, the choice of learning rate (η = 1e-4) may not have been most appropriate for this data set, as three of the models (out of 80) experienced *gradient explosion*. Gradient explosion led to overflows and NaN (not a number) losses that converted to Dice Coefficients of 0. Since the models are to be used for benchmarking, it would be fair to record the best performance that can be realistically obtained. As such, these three models were retrained. The results are available in Figure 5.6.

As seen in the figure, there are clear parallels with the results from training the GE data set benchmark models. Again, having less training data led to worse overall performance and greater variability. The same applied to model complexity, though the impact is mitigated with more training data. The models struggled with RV epicardium just as they did with the LV epicardium, likely for the same reasons. Surprisingly, the overall performance was not markedly worse than with GE data, even though RV is known for greater geometric complexity than LV. However, if the models that produced NaN losses due to gradient explosion were not retrained, there would be a significant impact on the results of U-Net 8 with 50 images for training (two models failed initially) and U-Net 16 with 100 images (one model failed initially).

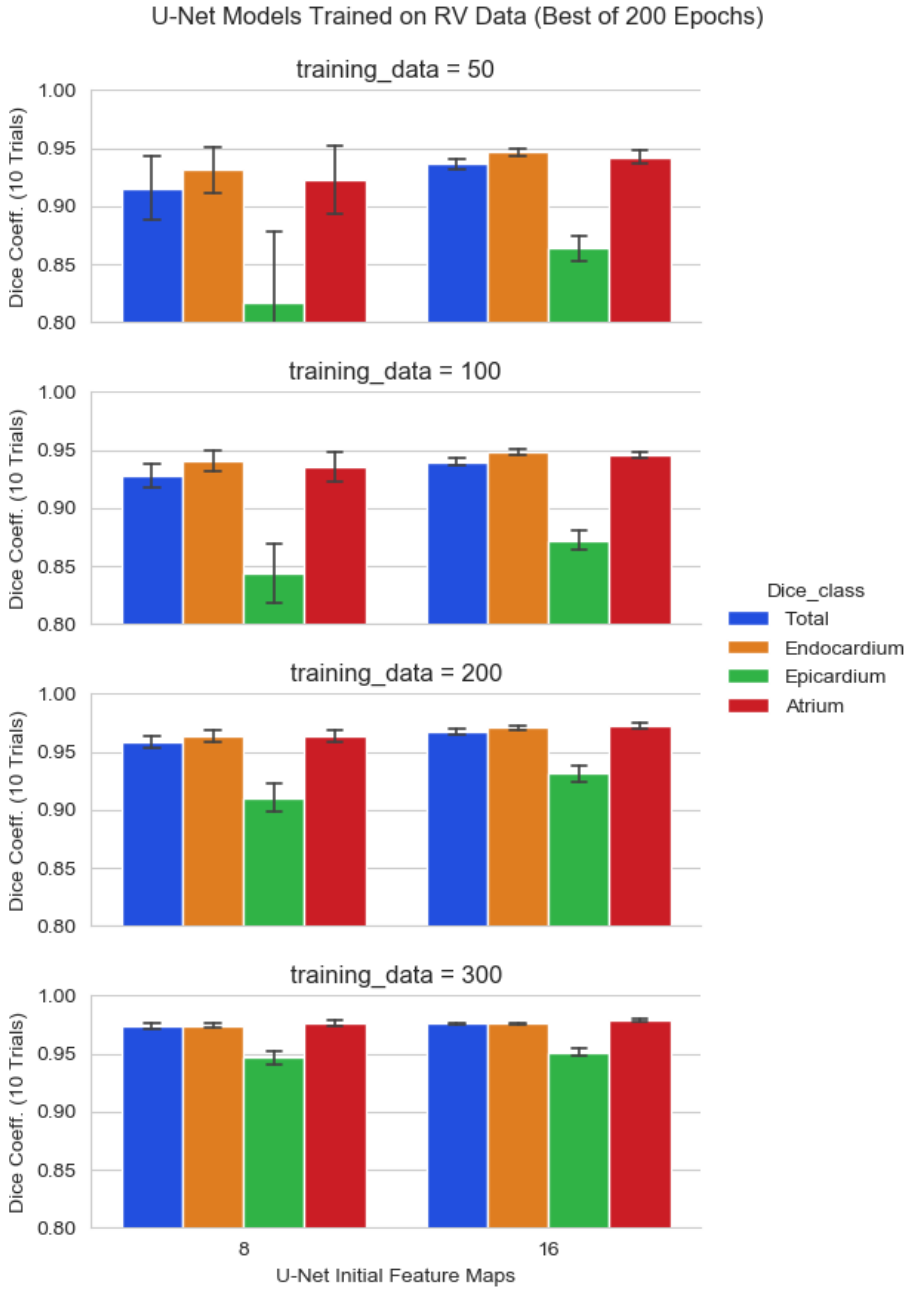


Figure 5.6: The Dice Coefficient values for benchmark U-Net models trained on right heart data (RV data set). The results are grouped by the number of initial feature maps and amount of training data. Two U-Net 8 models (on 50 images) and one U-Net 16 model (on 100 images) initially experienced gradient explosion and were retrained.

5.2.2 Transfer from CAMUS Data Set to RV Data Set

The models pre-trained on the CAMUS data were fine-tuned on the RV data (100 epochs of training, best performance recorded). The exact models used were, once more, U-Net 8 and U-Net 16, and the training subset was limited to 50, 100, 200, and 300 images. Again, 10 trials were performed with each parameter combination, and Figure 5.7 displays the results.

The initial concern about the models adjusting to the new task did not *fully* come to pass, and the results were comparable to the transfer between CAMUS and GE data. All models achieved nearly equal performance, even with low model complexity and little training data available. The largest increase occurred in the RV epicardium class - an already familiar pattern.

A direct comparison between the benchmark models and the fine-tuned models is provided in Figure 5.8. The figure confirms that the best relative increase in performance happened under the same parameters where the benchmark U-Net models struggled the most: 8 initial feature maps and 50 images available for training. The relative improvement for this parameter combination is even higher than for the CAMUS-GE transfer. Once more, as the amount of available training data increased, the differences evaporated.

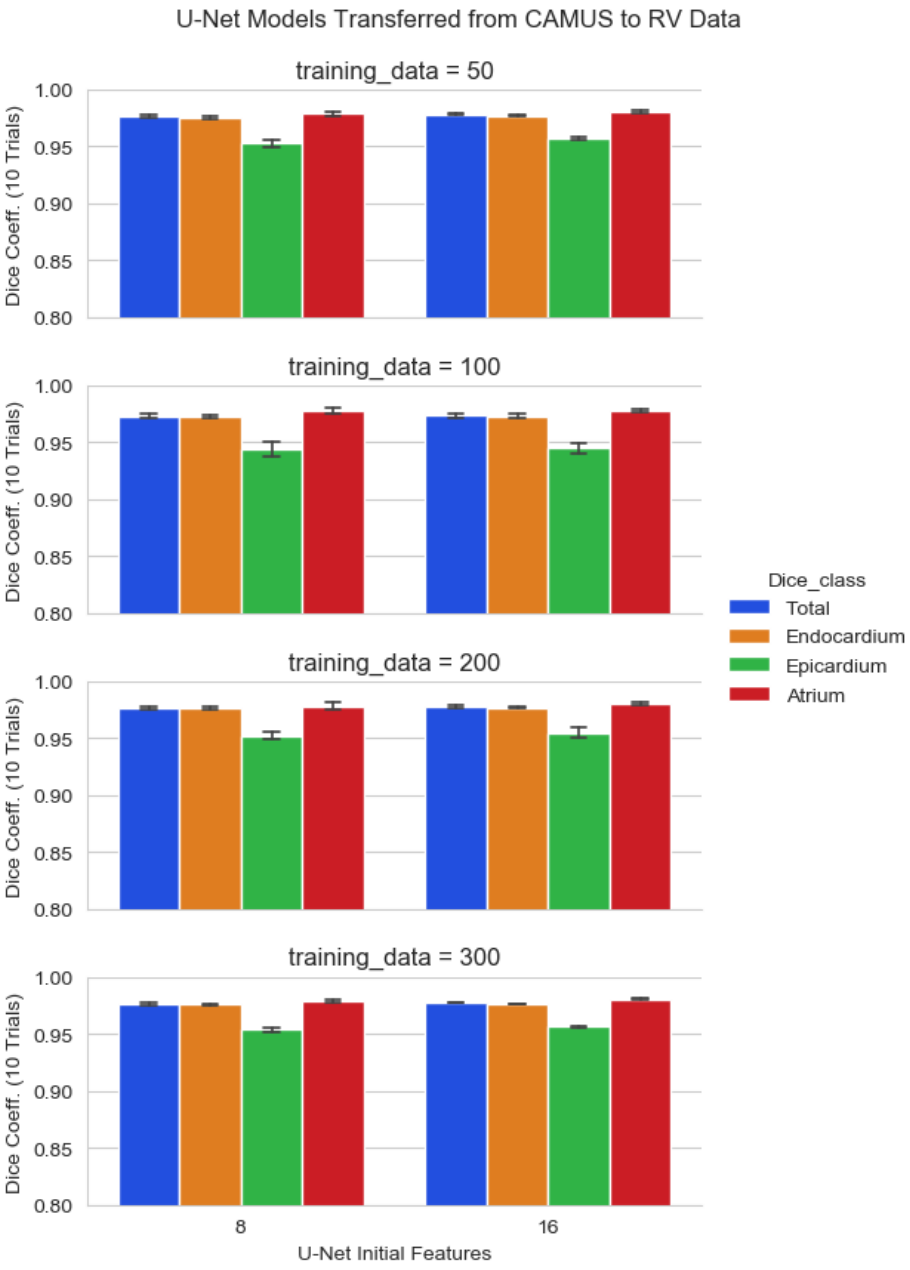


Figure 5.7: The Dice Coefficient values for U-Net models pre-trained on CAMUS data and then fine-tuned by further training on RV data set. The results are grouped by the number of initial feature maps and amount of training data.

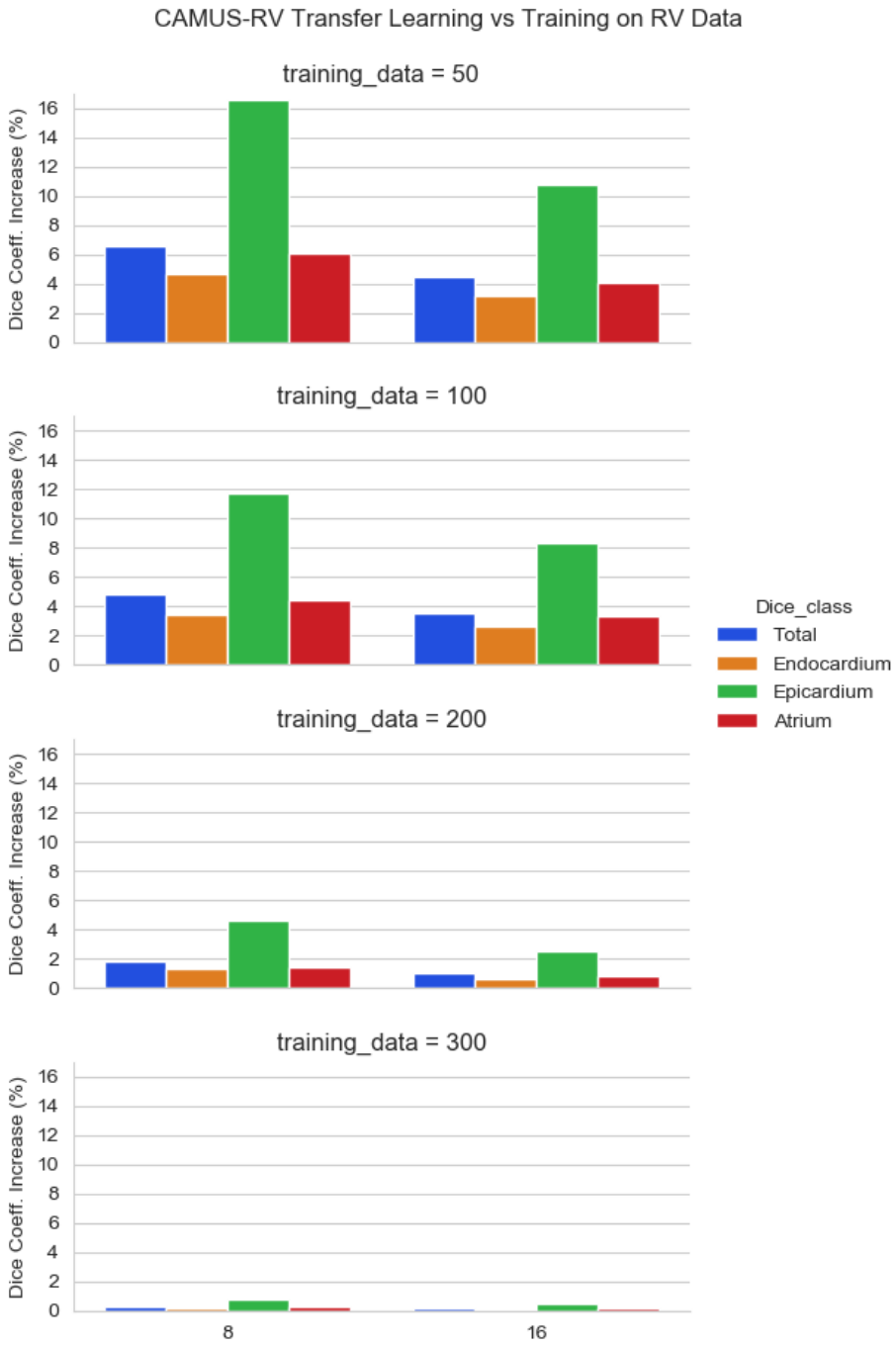


Figure 5.8: Comparison between Figures 5.6 and 5.7: the difference in the Dice Coefficient values (in percentage points) between the fine-tuned models and the benchmark models.

5.2.3 Transfer from GE Data Set to RV Data Set

One more transfer can be executed to double check the findings - the transfer from GE data set to RV data set. There is no need to train new models on the GE data set, as some of the old benchmark models can be reused (U-Net 8 and U-Net 16 models trained on 400 images). This recycling attempt may be seen as somewhat unfair, since the models had 200 epochs to train and not 100, but the experiment can still provide valuable insights. Likewise, the benchmark models for RV data set are already available from previous experiments. Similarly to the CAMUS-RV transfer, the models pre-trained on GE were fine-tuned on the RV data. Figure 5.9 offers the results, while Figure 5.10 provides the comparison to the benchmark models.

The figures demonstrate the same patterns as the CAMUS-RV transfer, but the models' performance is just slightly worse across the board. The most likely reason is that GE data set simply has a lot less data than CAMUS. The comparison of models that trained on 300 images serves as one case where transfer learning learning led to a slight decrease in performance. There is also barely any improvement with 200 images, especially for U-Net 16.

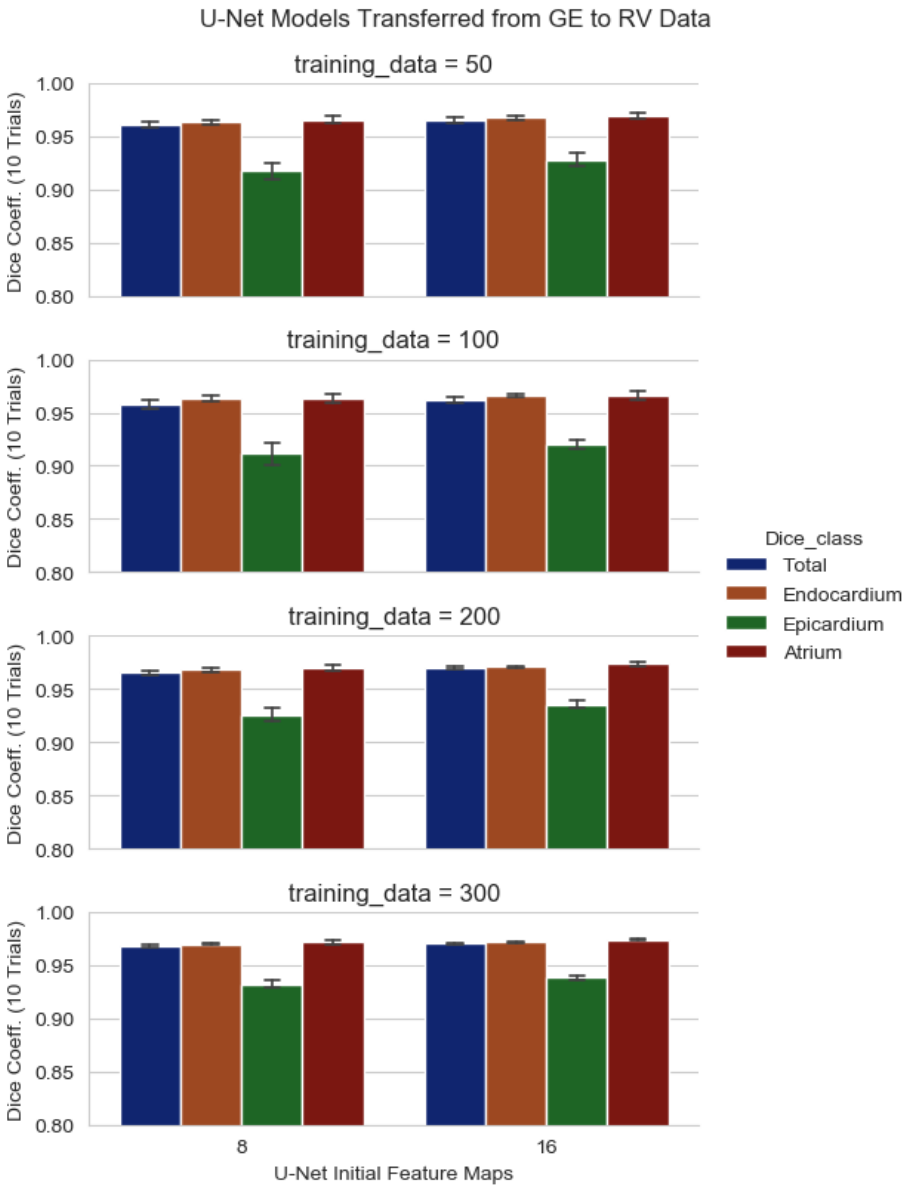


Figure 5.9: The Dice Coefficient values for U-Net models pre-trained on GE data and then fine-tuned by further training on RV data set. The results are grouped by the number of initial feature maps and amount of training data.

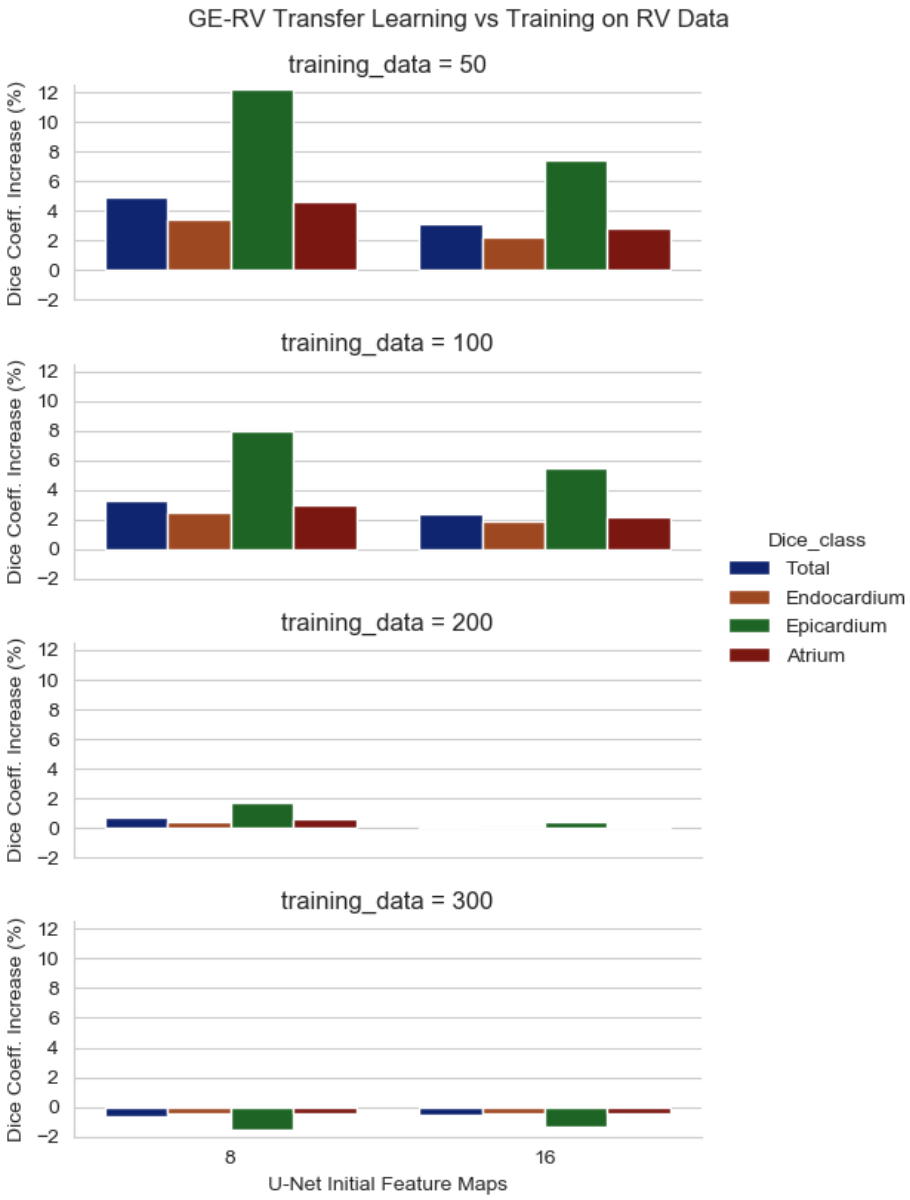


Figure 5.10: Comparison between Figures 5.6 and 5.9: the difference in the Dice Coefficient values (in percentage points) between the fine-tuned models and the benchmark models.

5.3 Applying Trained U-Net Models to Images from RV Data Set

As mentioned in Section 4.3, applying the U-Net models to individual echocardiograms may reveal additional information. Thus, predictions for the entire RV data set were created with two U-Net 8 models: one RV benchmark model trained on 50 images (one of the weakest), and the corresponding CAMUS-RV transfer model that used 50 images in fine-tuning (one of the strongest). Figure 5.11 offers the predictions of the first four images in RV data set along with the ground truth.

The first two images (001-ED and 001-ES) have the CAMUS-RV transfer model segment the chambers just slightly more accurately than the benchmark RV model, though even this much difference leads to a sharp increase in the Dice Coefficient values, especially for the epicardium (red). For the third image (002-ED), the RV benchmark model over-segmented the epicardium, but the transferred model did not. Similarly, the benchmark model under-segmented the atrium in 002-ES, but not the transferred model.

However, the CAMUS-RV model is not infallible, and does not always perform better. Figure 5.12 shows four more images from RV data set with corresponding predictions that are somewhat unusual.

For the first image (027-ES), the transferred model misplaced a part of the atrium, while the benchmark model performed well. For the other three images, the benchmark model "leaked" over to the left side of the heart, even though it was never trained to do so. It suggests that some of the features learned when segmenting one side of the heart are useful for the other side too. After all, the parts that "leaked" are not completely wrong, but rather incomplete. As for the transferred model, it was clearly eager to provide segmentation labels for both sides of the heart. Indeed, the model did not completely forget its previous task, which interfered with its performance to some degree.

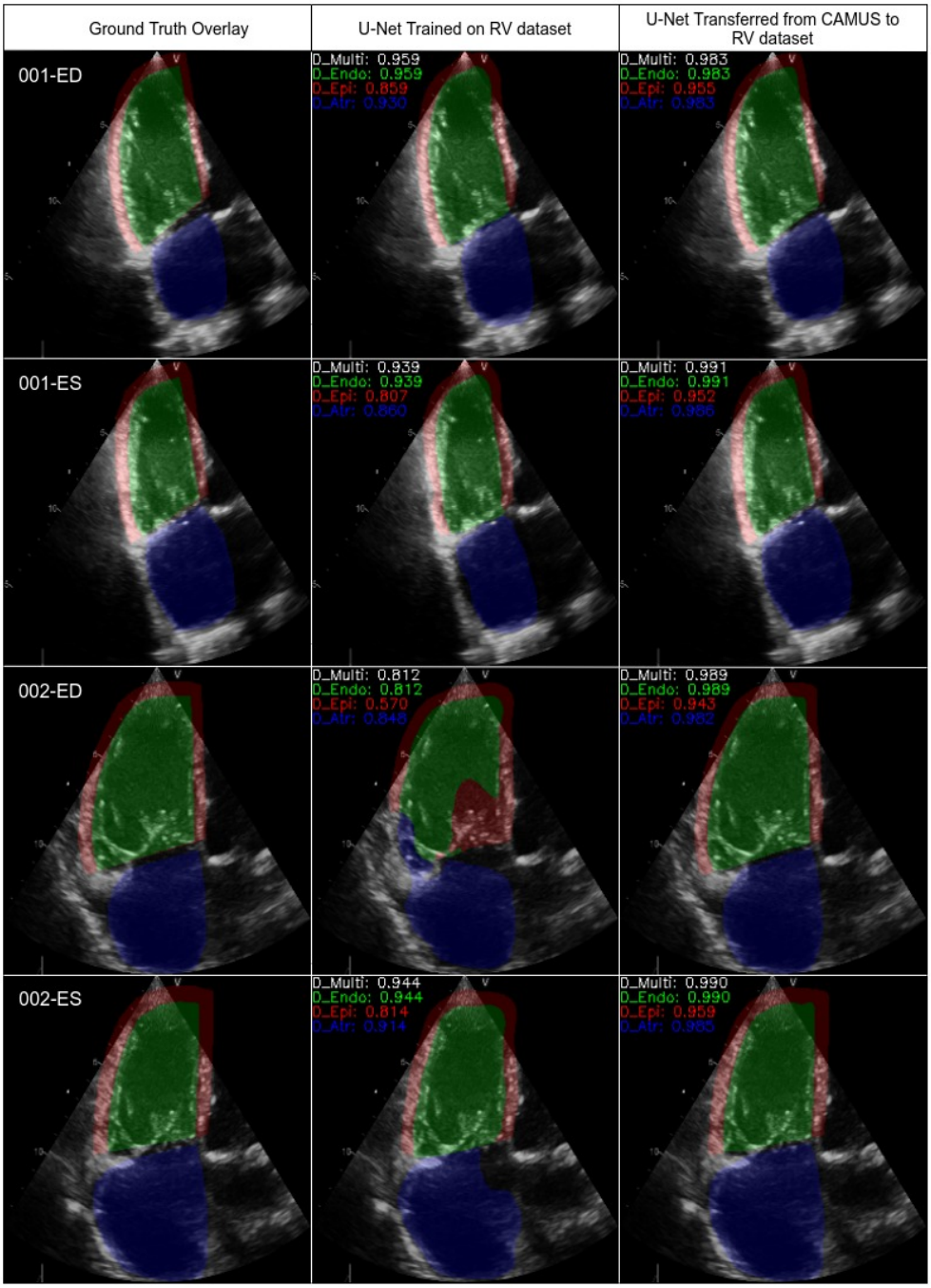


Figure 5.11: Ground truth and U-Net 8 predictions for the first four images in RV data set. The color codes are: green - endocardium, red - epicardium, blue - atrium. Dice Coefficients are provided for the predictions, but may be difficult to see.

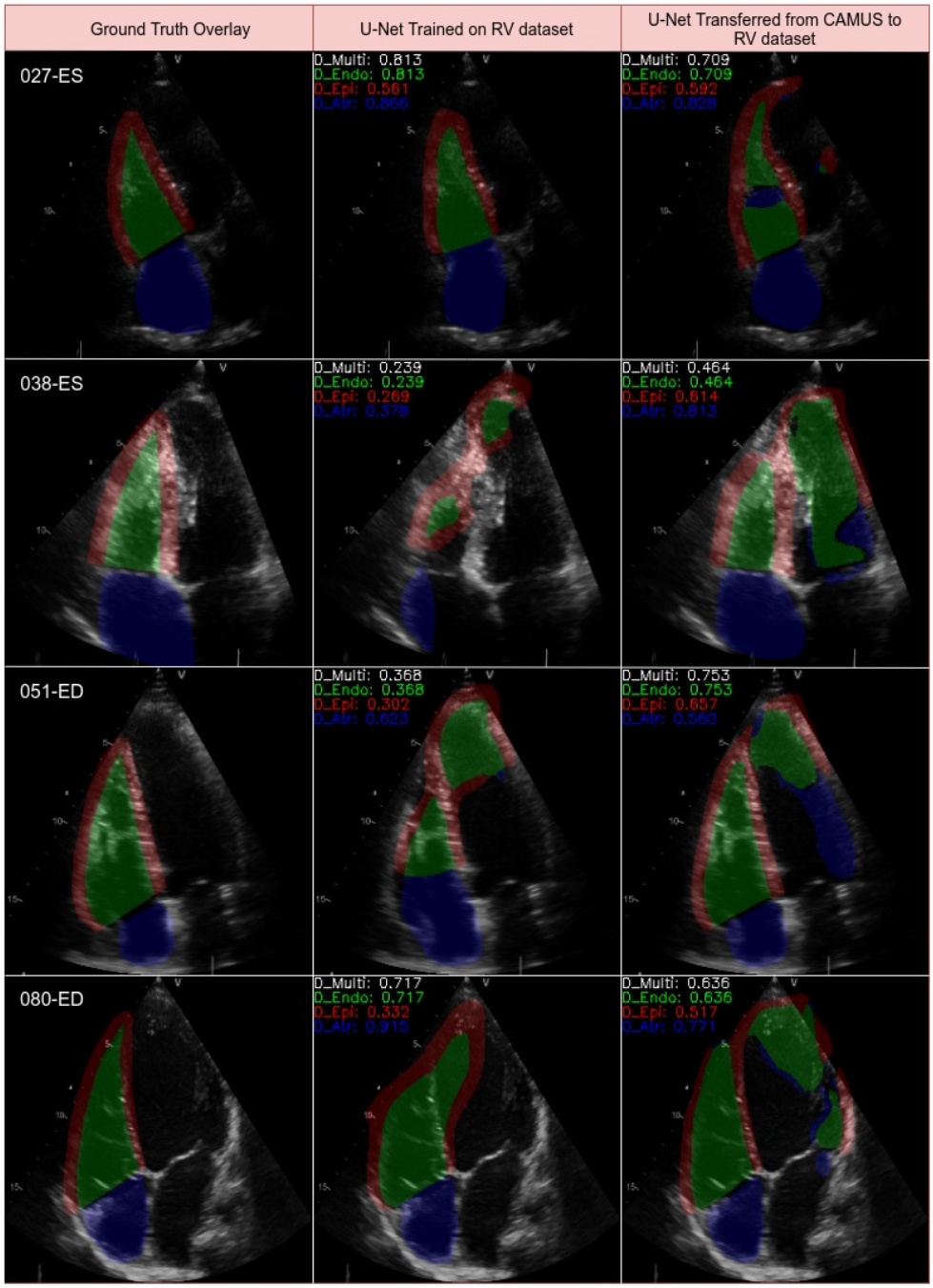


Figure 5.12: Ground truth and U-Net 8 predictions for another four images in RV data set. The color codes are: green - endocardium, red - epicardium, blue - atrium. Dice Coefficients are provided for the predictions, but may be difficult to see.

One may notice that Figure 5.11 is comprised of images that are more RV-focused, while Figure 5.12 mostly includes four-chamber views (with the possible exception of 027-ES). Reviewing other predictions also made it seem like the U-Net models struggled with four-chamber views, as previously theorized. These findings are in line with the research by *Genovese et al.*, where it is reported that quantification metrics extracted from RV-focused images are consistently larger, more reproducible, and less variable compared to measurements from four-chamber images [40]. The authors even recommend to only use RV-focused views for quantitative assessment of RV.

Unfortunately, it would be difficult to provide proper statistics of the models' performance on RV-focused vs four-chamber images. The RV data set is mixed, and it is not immediately clear which images are RV-focused and which are not. Some images include a large portion of LV and LA, but not a full view, possibly belonging to neither category.

Chapter 6

Discussion and Conclusion

This chapter summarizes the results of the experiments and forms the answer to the main research question (Section 6.1). In light of the new insights, the limitations of the work are revised, and possible directions for further research are proposed (Section 6.2).

6.1 Assessment of the Results

In the course of the work, a variety of transfer learning experiments with U-Net were conducted. As a trial run of the procedures, transfer learning involving CAMUS and GE data sets for segmentation of the left heart was attempted first. These experiments have shown that both the amount of training data and the complexity of U-Net contribute to segmentation performance, though there is a limit. It also became evident that U-Net may struggle with segmenting the epicardium in both ventricles, because the annotations were often imprecise for this region. Transfer of knowledge was effective in this scenario, with models that were initially weak benefiting the most - there was a 4% improvement in multi-class Dice Coefficient for these models. The Dice Coefficient for LV epicardium, the worst performing class, improved by over 10% for these models as well. The transfer brought all models to

approximately the same level of performance, just slightly above the best benchmark models. However, there was very little difference in performance between U-Net 16 and U-Net 32.

The overall setup had to be revised for further experiments. Since U-Net 32 appeared to be above the "complexity limit", it was removed from further procedures. Also, there was not enough data in the right heart data set (RV data set) to conduct the same range of the experiments. For this reason, the largest training subset was reduced from 400 to 300 images. The labels in RV data set were not binary like in CAMUS and GE data, and a decision had to be made about which parts still counted as parts of the label. All pixels with intensity above zero were included in the label, and it is theorized that over-segmentation could have taken place as a result.

Next, transfer learning that involves all data sets was attempted. CAMUS data set was larger than GE data set, while RV data set was the only one dedicated to the right heart chambers. Thus, the CAMUS-RV transfer was the main focus. Most of the patterns discovered during the CAMUS-GE transfer appeared once more. Despite having to "forget" their source task of segmenting the *left* heart in the four-chamber view images of RV data set, the U-Net models performed well in the transfer. Again, all models reached the level of performance slightly above the best benchmark models. **The weakest models improved their multi-class Dice Coefficient by more than 6% on average, while RV epicardium Dice Coefficient improved by 16%**. The CAMUS-RV transfer was thus even more successful than the CAMUS-GE transfer. It is worth noting that some of the RV benchmark models (3 out 80) had to be retrained due to the occurrence of gradient explosion. If these benchmark models were not retrained, the effectiveness of the transfer would look even higher than presented, but the comparison would hardly be fair.

In order to provide more data points, a transfer involving GE and RV data sets was performed. Here, some of the best GE benchmark

models were reused as pre-trained models. Most of the established patterns emerged during these experiments too. The weakest models improved by about 5% on the multi-class Dice Coefficient, and by 12% on the RV epicardium Dice Coefficient. However, the performance of the transferred models was slightly worse than that of the best benchmark models - a difference of about 0.5% in multi-class Dice Coefficient. For this arrangement, the data sets for the source and the target tasks consisted of 400 and 300 images, respectively. Overall, the results suggested that the ratio of source to target task data should be at least 2/1 or even 3/1 for transfer learning not to be a liability with these particular data sets.

The last of the experiments involved applying two U-Net 8 models to individual images in the RV data set. The models involved were an RV benchmark model trained on 50 images, and a CAMUS-RV transfer model that was fine-tuned on the same 50 images as well. These two models showcased the best scenario for transfer learning. A few examples were provided to demonstrate the higher precision of the CAMUS-RV transfer model. A few other examples involving a strange behavior from both models were shown as well. Here, the RV benchmark model demonstrated some awareness of LV endocardium and LV epicardium, despite never having been trained for it. This behavior suggested that segmentation of both the LV and the RV involves recognition of similar features. Predictions made by the CAMUS-RV transfer model revealed that it did not quite forget its source task, as it attempted to segment both the left heart and the right heart chambers in some cases. The models appeared to perform better on RV-focused images than four-chamber view images, but collecting proper statistics proved to be prohibitively difficult with the established setup.

The main research question can now be answered. **Transfer learning can be feasible for automated segmentation of the right heart chambers in 2D echocardiograms**. Of course, there are certain conditions: it may not be as useful with other algorithms as with

U-Net. One should also consider the amount of available data in both the source task and the target task data sets. With enough right heart data, there is likely no need to resort to transfer learning in the first place. However, if transfer learning is attempted, a lot of left heart data should be prepared: even two or three times the amount of right heart data may not be enough. In the example of the GE-RV transfer, the ratio of 8/1 (400 images for pre-training and 50 for fine-tuning) led to a significant improvement. The CAMUS-RV transfer has shown that even further improvement was possible with a ratio of over 20/1 (1,173 image for pre-training and 50 for fine-tuning). Note that this relationship does not necessarily hold if the amount of right heart data rises into the hundreds of images.

One may also wish to use only RV-focused images in the right heart data set. However, if four-chamber view images are not included there, removing them from the left heart data set may also be a reasonable option. Having only two chambers to consider at a time (LV/LA and RV/RA) may simplify the work for AI models, thus improving their performance in both pre-training and fine-tuning.

6.2 Possible Improvements and Further Research

The thesis answered the main research question ("**Is transfer learning *feasible* for automated segmentation of the right heart chambers in 2D echocardiograms?**"), but much more work can be done on the topic. Only a handful of parameter combinations were explored, with one algorithm, and only with the simplest transfer learning scheme. It is entirely possible that there is a combination of parameters and techniques that provides better results.

Given that the models' performance appeared to be different for RV-focused images compared to four-chamber view images, further work

with the data could be a reasonable idea. For example, the data sets could be split in two to separate the views. The process would be trivial for CAMUS data set with its wealth of additional information, but challenging for GE and RV data sets.

The setup for the experiments was not ideal, though that much had already been explained. There are no perfect stopping criteria for training, given that an AI model can suddenly improve after long periods of stagnation. It is therefore not obvious whether the way of comparing the benchmark models and the fine-tuned models was completely fair. Perhaps, information about when the models achieved their best performance should have been included with other figures, but some could see that information as excessive.

This work focused only on the task of segmentation, but it is just one subroutine within echocardiogram interpretation. Given more data, LV_{EF} and RV_{EF} could have been calculated for the images as an additional performance metric. Alternatively (or in addition), an algorithm for predicting ejection fraction could have been implemented, and transfer learning could be tested with that algorithm. A variety of other metrics could be included as well, but GE and RV data sets did not allow for that, as they only contained segmentation masks.

Still further outside is the process of echocardiogram acquisition - AI could be helpful for this task as well. However, it is a partly physical task, meaning that it would involve robotics. Considering the state of the art reviewed reviewed back in Section 2.4, the goal of AI-assisted image acquisition is not completely unrealistic. Robotic systems for medical ultrasound have already been designed by researchers, supporting remote acquisition [41]. Higher quality commercial robots for this task also seem to be available at this point. Furthermore, visual guidance algorithms based on Deep Learning for robots have also been proposed [42, 43]. A combination of these systems could be quite potent, bringing some degree of automation to echocardiographic examinations.

Bibliography

[1] *Burnout a Major and Growing Issue Among Nation's Cardiologists*. en. URL: https://www.cathlabdigest.com/content/burnout-major-and-growing-issue-among-nations-cardiologists (visited on 02/15/2021).

[2] M. Alsharqi et al. "Artificial intelligence and echocardiography". eng. In: *Echo Research and Practice* 5.4 (Dec. 2018), R115–R125. ISSN: 2055-0464. DOI: 10.1530/ERP-18-0056.

[3] Ashlee Davis et al. "Artificial Intelligence and Echocardiography: A Primer for Cardiac Sonographers". en. In: *Journal of the American Society of Echocardiography* 33.9 (Sept. 2020), pp. 1061–1066. ISSN: 0894-7317. DOI: 10.1016/j.echo.2020.04.025. URL: http://www.sciencedirect.com/science/article/pii/S089473172030256X (visited on 01/17/2021).

[4] Ellen Ostenfeld and Frank A Flachskampf. "Assessment of right ventricular volumes and ejection fraction by echocardiography: from geometric approximations to realistic shapes". In: *Echo Research and Practice* 2.1 (Mar. 2015), R1–R11. ISSN: 2055-0464. DOI: 10.1530/ERP-14-0077. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4676454/ (visited on 12/11/2020).

[5] Andrew Gilbert et al. "Generating Synthetic Labeled Data from Existing Anatomical Models: An Example with Echocardiography Segmentation". eng. In: *IEEE transactions on medical imag-*

ing PP (Jan. 2021). ISSN: 1558-254X. DOI: 10.1109/TMI.2021.3051806.

[6] S. Bierig and Anne Jones. "Accuracy and Cost Comparison of Ultrasound Versus Alternative Imaging Modalities, Including CT, MR, PET, and Angiography". In: *Journal of Diagnostic Medical Sonography* 25 (June 2009), pp. 138–144. DOI: 10.1177/8756479309336240.

[7] *Normal Heart Anatomy and Blood Flow - Pediatric Heart Specialists*. URL: https://pediatricheartspecialists.com/heart-education/14-normal/152-normal-heart-anatomy-and-blood-flow (visited on 02/23/2021).

[8] Michelle N. Berman, Connor Tupper, and Abhishek Bhardwaj. "Physiology, Left Ventricular Function". eng. In: *StatPearls*. Treasure Island (FL): StatPearls Publishing, 2020. URL: http://www.ncbi.nlm.nih.gov/books/NBK541098/ (visited on 12/11/2020).

[9] Ateet Kosaraju et al. "Left Ventricular Ejection Fraction". eng. In: *StatPearls*. Treasure Island (FL): StatPearls Publishing, 2020. URL: http://www.ncbi.nlm.nih.gov/books/NBK459131/ (visited on 12/11/2020).

[10] Theophilus E. Owan et al. "Trends in Prevalence and Outcome of Heart Failure with Preserved Ejection Fraction". In: *New England*
Journal of Medicine 355.3 (July 2006). Publisher: Massachusetts Medical Society _eprint: https://doi.org/10.1056/NEJMoa052256, pp. 251–259. ISSN: 0028-4793. DOI: 10.1056/NEJMoa052256. URL: https://doi.org/10.1056/NEJMoa052256 (visited on 12/21/2020).

[11] Barry A. Borlaug and Walter J. Paulus. "Heart failure with preserved ejection fraction: pathophysiology, diagnosis, and treatment". In: *European Heart Journal* 32.6 (Mar. 2011), pp. 670–

679. ISSN: 0195-668X. DOI: 10.1093/eurheartj/ehq426. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3056204/ (visited on 12/21/2020).

[12] Felipe Bisbal et al. "Atrial Failure as a Clinical Entity: JACC Review Topic of the Week". eng. In: *Journal of the American College of Cardiology* 75.2 (Jan. 2020), pp. 222–232. ISSN: 1558-3597. DOI: 10.1016/j.jacc.2019.11.013.

[13] Voelkel Norbert F. et al. "Right Ventricular Function and Failure". In: *Circulation* 114.17 (Oct. 2006). Publisher: American Heart Association, pp. 1883–1891. DOI: 10.1161/CIRCULATIONAHA.106.632208. URL: https://www.ahajournals.org/doi/10.1161/CIRCULATIONAHA.106.632208 (visited on 01/08/2021).

[14] Lawrence G. Rudski et al. "Guidelines for the Echocardiographic Assessment of the Right Heart in Adults: A Report from the American Society of Echocardiography". en. In: *Journal of the American Society of Echocardiography* 23.7 (July 2010), pp. 685–713. ISSN: 08947317. DOI: 10.1016/j.echo.2010.05.010. URL: https://linkinghub.elsevier.com/retrieve/pii/S0894731710004347 (visited on 12/21/2020).

[15] Richard A Harrigan and Kevin Jones. "Conditions affecting the right side of the heart". In: *BMJ : British Medical Journal* 324.7347 (May 2002), pp. 1201–1204. ISSN: 0959-8138. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1123164/ (visited on 01/12/2021).

[16] Geoff Strange et al. "Pulmonary hypertension: prevalence and mortality in the Armadale echocardiography cohort". en. In: *Heart* 98.24 (Dec. 2012). Publisher: BMJ Publishing Group Ltd Section: Original articles, pp. 1805–1811. ISSN: 1355-6037, 1468-201X. DOI: 10.1136/heartjnl-2012-301992. URL: https://heart.bmj.com/content/98/24/1805 (visited on 01/12/2021).

[17] *Cardiovascular diseases statistics - Statistics Explained*. URL: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Cardiovascular_diseases_statistics&oldid=261650 (visited on 01/12/2021).

[18] Thomas M. Gorter et al. "Right heart dysfunction and failure in heart failure with preserved ejection fraction: mechanisms and management. Position statement on behalf of the Heart Failure Association of the European Society of Cardiology". en. In: *European Journal of Heart Failure* 20.1 (2018). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/ejhf.1029, pp. 16–37. ISSN: 1879-0844. DOI: https://doi.org/10.1002/ejhf.1029. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ejhf.1029 (visited on 01/12/2021).

[19] John F. Park, Somanshu Banerjee, and Soban Umar. "In the eye of the storm: the right ventricle in COVID-19". In: *Pulmonary Circulation* 10.3 (July 2020). ISSN: 2045-8932. DOI: 10.1177/2045894020936660. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7333504/ (visited on 03/16/2021).

[20] Caroline Bleakley et al. "Right ventricular dysfunction in critically ill COVID-19 ARDS". English. In: *International Journal of Cardiology* 327 (Mar. 2021). Publisher: Elsevier, pp. 251–258. ISSN: 0167-5273, 1874-1754. DOI: 10.1016/j.ijcard.2020.11.043. URL: https://www.internationaljournalofcardiology.com/article/S0167-5273(20)34166-8/abstract (visited on 03/16/2021).

[21] Friedberg Mark K. and Redington Andrew N. "Right Versus Left Ventricular Failure". In: *Circulation* 129.9 (Mar. 2014). Publisher: American Heart Association, pp. 1033–1044. DOI: 10.1161/CIRCULATIONAHA.113.001375. URL: https://www.ahajournals.org/doi/full/10.1161/circulationaha.113.001375 (visited on 12/21/2020).

[22] James Moor. “The Dartmouth College Artificial Intelligence Conference: The Next Fifty Years”. en. In: *AI Magazine* 27.4 (Dec. 2006). Number: 4, pp. 87–87. ISSN: 2371-9621. DOI: 10.1609/aimag.v27i4.1911. URL: https://ojs.aaai.org/index.php/aimagazine/article/view/1911 (visited on 12/12/2020).

[23] Grace W. Lindsay. “Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future”. en. In: *Journal of Cognitive Neuroscience* (Feb. 2020), pp. 1–15. ISSN: 0898-929X, 1530-8898. DOI: 10.1162/jocn_a_01544. URL: https://www.mitpressjournals.org/doi/abs/10.1162/jocn_a_01544 (visited on 12/14/2020).

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. en. In: (), p. 9.

[25] Umut Güçlü and Marcel A. J. van Gerven. “Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations across the Ventral Stream”. en. In: *Journal of Neuroscience* 35.27 (July 2015). Publisher: Society for Neuroscience Section: Articles, pp. 10005–10014. ISSN: 0270-6474, 1529-2401. DOI: 10.1523/JNEUROSCI.5023-14.2015. URL: https://www.jneurosci.org/content/35/27/10005 (visited on 12/14/2020).

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *arXiv:1505.04597 [cs]* (May 2015). arXiv: 1505.04597. URL: http://arxiv.org/abs/1505.04597 (visited on 01/22/2021).

[27] Minyoung Huh, Pulkit Agrawal, and Alexei A. Efros. “What makes ImageNet good for transfer learning?” en. In: *arXiv:1608.08614 [cs]* (Dec. 2016). arXiv: 1608.08614. URL: http://arxiv.org/abs/1608.08614 (visited on 12/20/2020).

[28] Yunhui Guo et al. “SpotTune: Transfer Learning through Adaptive Fine-tuning”. In: *arXiv:1811.08737 [cs, stat]* (Nov. 2018). arXiv: 1811.08737. URL: http://arxiv.org/abs/1811.08737 (visited on 02/23/2021).

[29] Jeffrey Zhang et al. “Fully Automated Echocardiogram Interpretation in Clinical Practice”. eng. In: *Circulation* 138.16 (2018), pp. 1623–1635. ISSN: 1524-4539. DOI: 10.1161/CIRCULATIONAHA.118.034338.

[30] E. Smistad et al. “Fully Automatic Real-Time Ejection Fraction and MAPSE Measurements in 2D Echocardiography Using Deep Neural Networks”. In: *2018 IEEE International Ultrasonics Symposium (IUS)*. ISSN: 1948-5727. Oct. 2018, pp. 1–4. DOI: 10.1109/ULTSYM.2018.8579886.

[31] Roberto M. Lang et al. “Recommendations for Cardiac Chamber Quantification by Echocardiography in Adults: An Update from the American Society of Echocardiography and the European Association of Cardiovascular Imaging”. en. In: *Journal of the American Society of Echocardiography* 28.1 (Jan. 2015), 1–39.e14. ISSN: 08947317. DOI: 10.1016/j.echo.2014.10.003. URL: https://linkinghub.elsevier.com/retrieve/pii/S0894731714007457 (visited on 01/03/2021).

[32] Amirata Ghorbani et al. “Deep learning interpretation of echocardiograms”. en. In: *npj Digital Medicine* 3.1 (Jan. 2020). Number: 1 Publisher: Nature Publishing Group, pp. 1–10. ISSN: 2398-6352. DOI: 10.1038/s41746-019-0216-8. URL: https://www.nature.com/articles/s41746-019-0216-8 (visited on 01/03/2021).

[33] David Ouyang et al. “Video-based AI for beat-to-beat assessment of cardiac function”. en. In: *Nature* 580.7802 (Apr. 2020), pp. 252–256. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-020-2145-8. URL: http://www.nature.com/articles/s41586-020-2145-8 (visited on 02/03/2021).

[34] A Karuzas et al. “544 Deep learning in segmentation and function evaluation of right ventricle in 2D echocardiography”. In: *European Heart Journal - Cardiovascular Imaging* 21.jez319.278 (Jan. 2020). ISSN: 2047-2404. DOI: 10.1093/ehjci/jez319.278. URL: https://doi.org/10.1093/ehjci/jez319.278 (visited on 01/04/2021).

[35] Ashley N. Beecy et al. “Development of novel machine learning model for right ventricular quantification on echocardiography—A multimodality validation study”. en. In: *Echocardiography* 37.5 (2020). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/echo.14674, pp. 688–697. ISSN: 1540-8175. DOI: https://doi.org/10.1111/echo.14674. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/echo.14674 (visited on 01/04/2021).

[36] Sarah Leclerc et al. “Deep Learning for Segmentation Using an Open Large-Scale Dataset in 2D Echocardiography”. eng. In: *IEEE transactions on medical imaging* 38.9 (Sept. 2019), pp. 2198–2210. ISSN: 1558-254X. DOI: 10.1109/TMI.2019.2900516.

[37] Pfaffenberger Stefan et al. “Size Matters! Impact of Age, Sex, Height, and Weight on the Normal Heart Size”. In: *Circulation: Cardiovascular Imaging* 6.6 (Nov. 2013). Publisher: American Heart Association, pp. 1073–1079. DOI: 10.1161/CIRCIMAGING.113.000690. URL: https://www.ahajournals.org/doi/full/10.1161/CIRCIMAGING.113.000690 (visited on 12/27/2020).

[38] Sergey Ioffe and Christian Szegedy. “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In: *arXiv:1502.03167 [cs]* (Mar. 2015). arXiv: 1502.03167. URL: http://arxiv.org/abs/1502.03167 (visited on 02/06/2021).

[39] Intisar Rizwan I Haque and Jeremiah Neubert. “Deep learning approaches to biomedical image segmentation”. en. In: *Informatics in Medicine Unlocked* 18 (Jan. 2020), p. 100297. ISSN: 2352-9148. DOI: 10.1016/j.imu.2020.100297. URL: https://www.sciencedirect.com/science/article/pii/S235291481930214X (visited on 02/08/2021).

[40] Davide Genovese et al. “Comparison Between Four-Chamber and Right Ventricular-Focused Views for the Quantitative Evaluation of Right Ventricular Size and Function”. eng. In: *Journal of the American Society of Echocardiography: Official Publication of the American Society of Echocardiography* 32.4 (Apr. 2019), pp. 484–494. ISSN: 1097-6795. DOI: 10.1016/j.echo.2018.11.014.

[41] Kim Mathiassen et al. “An Ultrasound Robotic System Using the Commercial Robot UR5”. English. In: *Frontiers in Robotics and AI* 3 (2016). Publisher: Frontiers. ISSN: 2296-9144. DOI: 10.3389/frobt.2016.00001. URL: https://www.frontiersin.org/articles/10.3389/frobt.2016.00001/full (visited on 02/15/2021).

[42] Aseem Saxena et al. “Exploring Convolutional Networks for End-to-End Visual Servoing”. en. In: *arXiv:1706.03220 [cs]* (June 2017). arXiv: 1706.03220. URL: http://arxiv.org/abs/1706.03220 (visited on 02/10/2021).

[43] Quentin Bateux et al. “Training Deep Neural Networks for Visual Servoing”. en. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, QLD: IEEE, May 2018, pp. 3307–3314. ISBN: 978-1-5386-3081-5. DOI: 10.1109/ICRA.2018.8461068. URL: https://ieeexplore.ieee.org/document/8461068/ (visited on 02/10/2021).