Automated Echocardiographic Image Interpretation Using Artificial Intelligence

UNIVERSITY OF
LINCOLN

Neda Azarmehr

School of Computer Science

College of Science

University of Lincoln

Submitted in partial satisfaction of the requirements for the
Degree of Doctor of Philosophy
in Computer Science

Supervisors Prof. Xujiong Ye, Prof. Massoud Zolgharni

March 2021

Acknowledgements

I have received invaluable support, encouragement, and guidance from many people during this journey. The enthusiasm of colleagues, family, and friends towards my research was a vital source of motivation in those moments when there was not forward or inverse problem to be solved. I would like to express my gratitude to everybody who has supported me during this momentous phase in my life. I would never have been able to do this without you.

The support and guidance of my supervisor prof Xujiong Ye and prof Massoud Zolgharni have been key to my success in completing this thesis. They have been a consistent source of guidance. Their critical and constructive advice throughout the project has been significant to the progress and quality of this research. I would like to express my deepest sense of gratitude to them for the many opportunities they have offered me to improve my academic profile.

I would also like to extend my gratitude to my third supervisor Dr. Faraz Janan. I would also like to express my sincere thanks to Professor Luc Bidaut for providing computing resources. I would also like to thank Professor Darrel Francis at Imperial College London for supporting and providing the datasets.

Helpful suggestions by the PhD examiners, Professor Hui Wang from the University of Ulster and Dr James Brown from the University of Lincoln, is appreciated.

The financial support from the University of Lincoln, School of Computer Science gratefully acknowledged. Furthermore, I express my gratitude to the University of Lincoln staff for their cooperation.

I would like to thank my loving family who has always encouraged and supported my efforts. I can never say enough the boundless support, love, and guidance that

my parents have provided me since the day I was born till today. I dedicate my thesis to my parents for their endless love, support, encouragement, and sacrifices.

Abstract

In addition to remaining as one of the leading causes of global mortality, cardiovascular disease has a significant impact on overall health, well-being, and life expectancy. Therefore, early detection of anomalies in cardiac function has become essential for early treatment, and therefore reduction in mortalities. Echocardiography is the most commonly used modality for evaluating the structure and function of the heart. Analysis of echocardiographic images has an important role in the clinical practice in assessing the cardiac morphology and function and thereby reaching a diagnosis.

The process of interpretation of echocardiographic images is considered challenging for several reasons. The manual annotation is still a daily work in the clinical routine due to the lack of reliable automatic interpretation methods. This can lead to time-consuming tasks that are prone to intra- and inter-observer variability. Echocardiographic images inherently suffer from a high level of noise and poor qualities. Therefore, although several studies have attempted automating the process, this remains a challenging task, and improving the accuracy of automatic echocardiography interpretation is an ongoing field.

Advances in Artificial Intelligence and Deep Learning can help to construct an automated, scalable pipeline for echocardiographic image interpretation steps, including view classification, phase-detection, image segmentation with a focus on border detection, quantification of structure, and measurement of the clinical markers. This thesis aims to develop optimised automated methods for the three individual steps forming part of an echocardiographic exam, namely view classification, left ventricle segmentation, quantification, and measurement of left ventricle structure. Various Neural Architecture Search methods were employed to design efficient neural network architectures for the above tasks. Finally, an optimisation-based speckle tracking

echocardiography algorithm was proposed to estimate the myocardial tissue velocities and cardiac deformation. The algorithm was adopted to measure cardiac strain which is used for detecting myocardial ischaemia.

All proposed techniques were compared with the existing state-of-the-art methods. To this end, publicly available patients datasets, as well as two private datasets provided by the clinical partners to this project, were used for developments and comprehensive performance evaluations of the proposed techniques. Results demonstrated the feasibility of using automated tools for reliable echocardiographic image interpretations, which can be used as assistive tools to clinicians in obtaining clinical measurements.

Table of Contents

List of Figures

List of Tables

Acronyms

2D Two Dimensional

3D Three Dimensional

A2C Apical Two-Chamber

A4C Apical Four-Chamber

AI Artificial Intelligence

ASE American Society of Echocardiography

ASPP Atrous Spatial Pyramid Pooling

AVC Aortic Valve Closure

BHF British Heart Foundation

BM Block Matching

Cardiac CT Cardiac Computed Tomography

CNN Convolutional Neural Network

DARTS Differentiable Architecture Search

DBN Deep Belief Network

DC Dice Coefficient

EACVI European Association of Cardiovascular Imaging

ECG Electrocardiogram

ED End-Diastolic

EF Ejection Fraction

ENAS Efficient Neural Architecture Search

ES End-Systolic

FC Fully Connected

FCN Fully Convolution Network

GLS Global Longitudinal Strain

GT Ground-Truth

HD Hausdorff Distance

IoU Intersection-Over-Union

KF Kalman Filter

LSTM Long short-term memory

LV Left Ventricle

MRI Magnetic Resonance Imaging

MVC Mitral Valve Closure

NAS Neural Architecture Search

PACS Picture Archiving and Communication Systems

ReLU Rectified Linear Unit

RL Reinforcement Learning

RNN Recurrent Neural Network

RV Right Ventricle

SGD Stochastic Gradient Descent

SSD Sum of Squared Differences

STE Speckle Tracking Echocardiography

TD Transitions Down

TDI Tissue Doppler Imaging

TEE Transesophageal Echocardiogram

TTE Transthoracic Echocardiography

TU Transitions Up

Chapter 1

Introduction

In 2016, of the 56.9 million deaths worldwide, the highest proportion can be attributed to ischaemic heart disease and stroke; accounting for 15.2 million deaths combined. These diseases have remained the leading cause of global mortality in the last 15 years (World Health Organisation, 2018). Therefore, early detection of anomalies in cardiac function has become crucial for effective treatment and therefore reduction in fatalities (Konofagou et al., 2011). Evaluation of cardiac motion plays a significant role in the quantification of cardiac muscle elasticity and contractility, which can help to distinguish between abnormal and normal cardiac function.

Medical image processing plays a significant contribution to clinical procedures by providing clinicians with automated tools to diagnose and treat cardiovascular disease. Among different imaging modalities, echocardiography is commonly used in both clinical procedures and research as a non-invasive technique to evaluate the structure and function of the heart (Nikita, 2013).

Advances in Artificial Intelligence (AI) can help to develop an automated, scalable pipeline for echocardiographic image analysis (displayed in Figure 1.1), including: (a) view classification (identification), (b) phase-detection (c) image segmentation, (d) and clinical measurements. Further details are provided in the following section.

In this chapter, an overview of echocardiography image interpretation, the problem statement for automating the procedure is presented. Furthermore, the motivation, main aim and objectives, and the contributions of this research are provided. Finally, the thesis outline and research consortium are introduced.

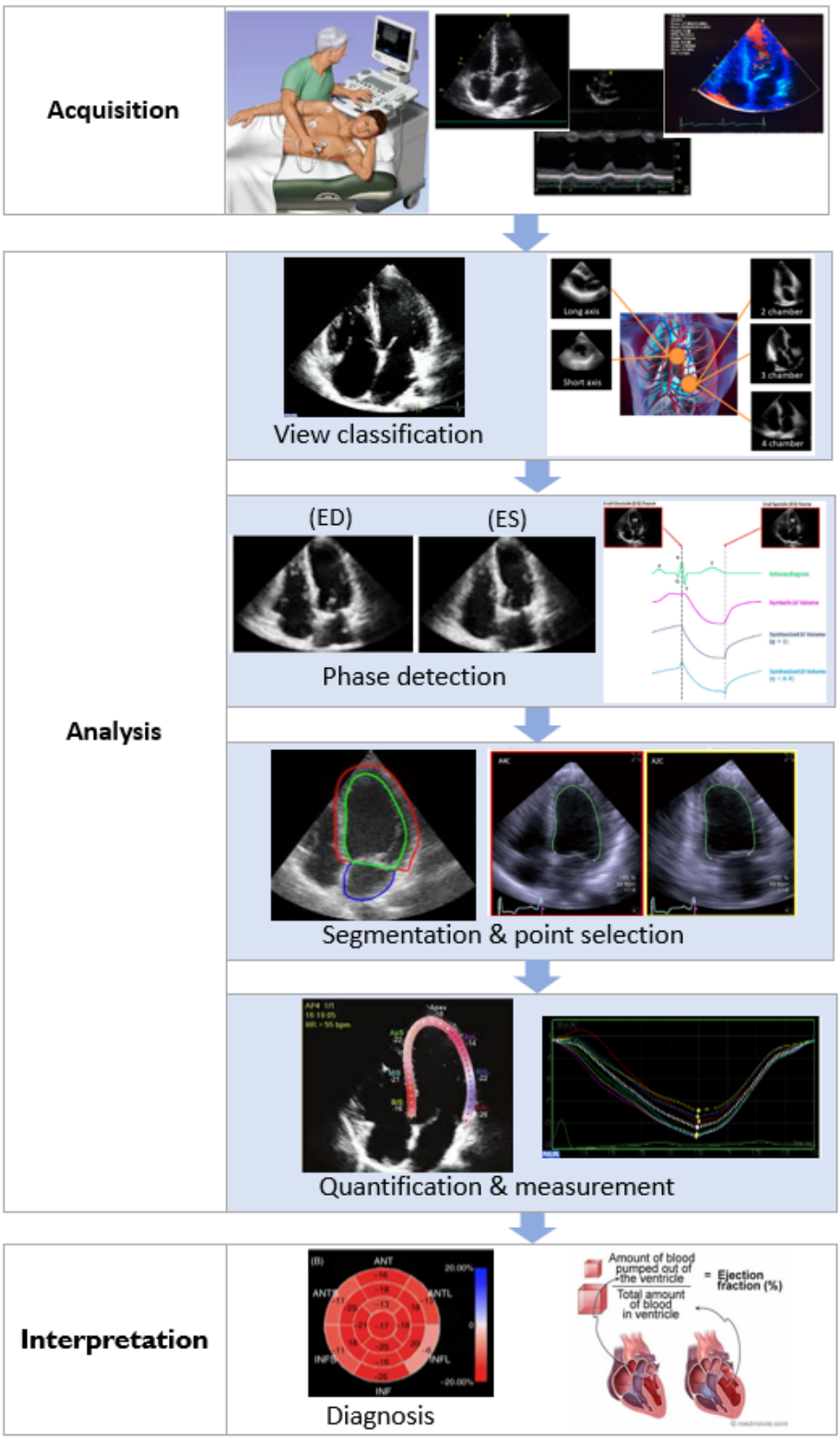


Figure 1.1: Process of an echocardiographic exam, which is currently an entirely manual process. In the envisaged automated pipeline, while the image acquisition and interpretation will still be carried out by a human operator, different steps in image analysis can potentially be automated.

1.1 Clinical Context and Problem Statement

Since many cardiac pathologies result in local myocardial dysfunction, the examination of local wall motion and deformation has gained significant attention over the past decade. Also, several studies have proven their accuracy and consistency (Mondillo et al., 2011; Geyer et al., 2010; Voigt et al., 2015). The focus of this thesis is upon cardiac ultrasound images (i.e., echocardiography) since they possess high temporal resolution, low cost, good compatibility, and extensive availability. However, the processing of ultrasound images can present significant difficulty due to the typically high level of noise to signal ratio found in them. Additionally, in cardiac ultrasound images, tracking walls of the heart is problematic due to the lower resolutions in the lateral wall of the heart, and the nature of heart motion (Golemati, Gastounioti and Nikita, 2016). Below is an overview of the several steps involved in echocardiography image interpretation.

1.1.1 Echocardiography View Classification

Echocardiography examinations are typically focused upon protocols containing diverse probe postures, providing various views of the heart anatomy. Standard echocardiographic views require imaging the heart from multiple windows (Lang, Badano et al., 2015). Each window is specified by the transducer position that will be explained in more detail in Chapter 2.

The interpretation of echocardiography images begins with view identification (i.e., classifying each acquired image to the corresponding cardiac views) which is currently done manually in a laborious process.

The appearance of images acquired in the same view of heart could be different for each patient due to two reasons: i) depending on the physical characteristics of patients, their heart structure could slightly vary, ii) since there is no specific marker area to place the transducer on the patient body, the appearance-based method cannot be applied for the view classification issue (Balaji, Subashini and Chidambaram, 2015). Therefore, accurate automatic classification of heart views has the poten-

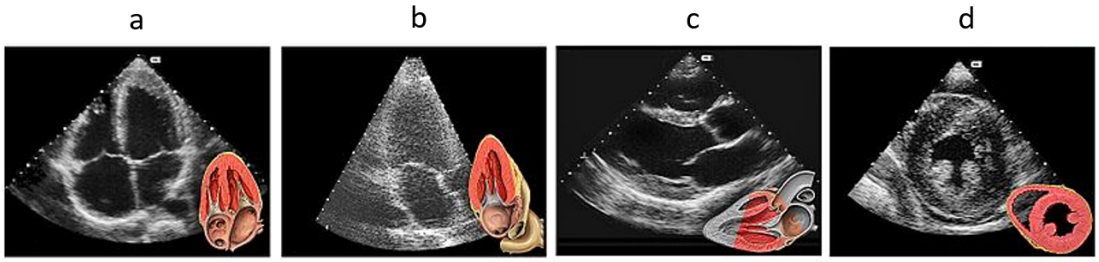


Figure 1.2: Examples of cardiac views in transthoracic echocardiography: a: apical four-chamber left ventricle focused (A4CH-LV), b: apical two-chamber (A2CH), c:parasternal long-axis (PLAX-Full), d: parasternal short-axis left ventricle focused (PSAX-LV), figure recreated from (Lynch and Jaffe, 2006).

tial to streamline workflow by aiding echocardiographers in reducing the inter-user discrepancy, improving the accuracy for high throughput of echocardiographic data, and the subsequent diagnosis.

Moreover, as is frequently observed in clinical practice, images from different modalities are managed and stored in Picture Archiving and Communication Systems (PACS). Recently, echocardiographic software packages, such as EchoPAC [1] and QLAB (Philips) [2], attempt automating the tasks during the image analysis. However, they still require some level of human involvement in detecting relevant views. Furthermore, echocardiography image frames are not easily discernible by the operator in addition to high levels of background noise. Therefore, automatic view classification could be crucially beneficial for pre-labelling a large database of unclassified images (Khamis, Zurakhov et al., 2017). An example of different echo views is outlined in Figure 1.2.

Automation of cardiac view detection will be discussed in Chapter 4.

1.1.2 Phase Detection

Cardiac function/motion is a cyclic process, and echocardiographic measurements in an image sequence usually relate to certain time points in the cycle, namely End-Diastolic (ED) and End-Systolic (ES) frames (Hall, 2016; Fukuta and Little,

[1] https://www.gehealthcare.co.uk/products/ultrasound/vivid/echopac
[2] https://www.philips.co.uk/healthcare/product/HCNOCTN14

2008). The frame after ED is considered as systolic phase, i.e., the beginning of a new cardiac cycle (heartbeat), and is typically marked by Mitral Valve Closure (MVC). Conversely, the (ES) frame is used to describe the end of systole (beginning of diastole) and is marked by Aortic Valve Closure (AVC). Mada et al. (2015) illustrated the importance of accurate identification of ED and ES frames (Mada et al., 2015). An error of just two to three frames in detecting ES frames causes approximately a 10% difference in segmental ES strain, which is one of the important clinical markers. Additionally, the consequence of incorrect identification of ED and ES frames can be extensive; impairing concordance between observers in both research and clinical practice (Amundsen, 2015). Consequently, automated methods for the accurate ED and ES phase-detection could significantly contribute to improving the consistency of echocardiographic quantification.

Automation of cardiac phase detection was carried out in our research group, to which I contributed [3].

1.1.3 Left Ventricle Segmentation

Image segmentation is a procedure that delineates the boundaries of an object of interest within an image to simplify the representation of an image into something more meaningful and easier to analyse. In the interpretation of echocardiography images, one of the main steps is cardiac image segmentation which partitions the image into several semantically (i.e., anatomically) meaningful regions that allow extracting the quantitative measures such as the myocardial mass, wall thickness, Left Ventricle (LV) and Right Ventricle (RV) volume as well as Ejection Fraction (EF) (C. Chen et al., 2020).

The quantification of the LV shape and deformation relies on the accurate segmentation of the LV contour in ED and ES frames (Raynaud et al., 2017). In the context of echocardiography interpretation, this procedure is often performed manually. However, a computer can also be taught to determine the same structure (Deo et al., 2017). At present, the manual segmentation of the LV suffers from various complic-

[3] https://github.com/intsav/EchoPhaseDetection

ations such as needs to be carried out only by an experienced clinician, inevitable inter- and intra-observer variability in the annotations, and it is laborious and must be repeated for each patient.

Therefore, accurate automatic segmentation of the LV has the potential to simplify workflow by aiding echocardiographers in reducing the inter-user discrepancy, thereby improving the accuracy for high throughput of echocardiographic data and subsequent diagnosis. Nevertheless, segmentation of LV is a challenging task that has to handle some problems inherent in ultrasound imaging such as, low signal-noise ratio, edge dropout, and artifacts (Suyu Dong, G. Luo, Sun et al., 2016).

Automation of the LV segmentation will be discussed in Chapter 5.

1.1.4 Quantification and Strain Measurement

Speckle Tracking Echocardiography (STE) is considered to be one of the most commonly used techniques for global and regional quantitative assessment of myocardial function (Mondillo et al., 2011). Speckle tracking is based on analysis of the spatial dislocation of speckles (i.e., spots generated by the interaction between the ultrasound beam and myocardial tissue) on image sequences (Mondillo et al., 2011; C. B. R. Liberato et al., 2020). Displacement of these speckles is due to myocardial deformation from which myocardial strain is calculated as a mechanical measure, and will be discussed further in Chapter 2.

Before introducing the STE, Tagged Magnetic Resonance Imaging (MRI) was the only imaging modality used for measuring the myocardial strain. However, the use of tagged MRI is limited because of its poor availability, cost-intensive, time-consuming image analysis, and complex acquisitions (Mondillo et al., 2011).

Although commercial STE software packages exist, the measurements they provide yield unsatisfactorily wide discrepancies between measurements on the same patient. To address this issue, the European Association of Cardiovascular Imaging (EACVI) and the American Society of Echocardiography (ASE), along with representatives from all vendors, have been endorsing a "task force" aimed to reduce the inter-vendor variability of strain measurement. They propose acceptance in the clinical practice of

inter-vendor variability up to 10% (Voigt et al., 2014; James D Thomas and Badano, 2013). However, currently used software packages have variability exceeding 10%. Several approaches for speckle tracking in ultrasound sequences have been proposed. However, it is a complicated task in which can be improved (Tavakoli et al., 2008; Z. Liu and J. Luo, 2017; Garcia, Lantelme and Saloux, 2018; Bahreini Toosi, Zarghani, Poorzand et al., 2019).

Therefore, the development of fully automated reliable, and reproducible STE techniques is highly desirable and will be discussed in Chapter 6 and 7.

1.2 Motivation

Medical diagnosis and treatment tasks using computer-aided systems is a fast-growing field of research that assists the clinician in obtaining measurements and identifying anomalies with more accuracy, precision, and greater speed. Medical image processing has an important influence in clinical procedures such as cardiac analysis focuses on ultrasound image modality which provides clinicians automated tools to support diagnosis and treatment tasks.

As discussed previously, in order to assess the cardiac function in ultrasound images, accurate view identification and LV segmentation may help to reduce inter-user discrepancy and provide fast and accurate measurement for high throughput of data. However, view identification and segmentation of the LV are very challenging tasks due to the high variation in shape, low image quality, and high level of noise that exist in echocardiography images. Several classifications and segmentation models have been proposed which will be reviewed in Chapter 4 and 5. However, there is potential for future improvements in terms of the reliability and accuracy of the techniques.

In recent years, deep learning has successfully been applied to the automated analysis of medical images including, tasks such as image classification, detection, and segmentation. The performance of the deep learning models is dependent on the configuration of the Convolutional Neural Network (CNN) architectures employed.

However, the manual design of a CNN architecture is a time-consuming and error-prone process. Therefore, it is also necessary to develop networks capable of automating this complex process. This project addresses the need for automated neural architecture design by leveraging recent algorithmic developments known as Neural Architecture Search (NAS) (H. Liu, Simonyan and Yang, 2019).

Echocardiographic techniques, such as strain imaging, have emerged as promising quantitative tools in measuring LV function with superior prognostic value to EF for predicting adverse cardiac events (Kalam, Otahal and Thomas H Marwick, 2014). Clinical feasibility of strain resulting from STE has been demonstrated in many studies (Barbosa et al., 2014; Ferraiuoli et al., 2019; Rodriguez et al., 2014; Joos et al., 2018; Hui and Xinhua, 2020) which will be discussed further in Chapter 6.

For example, strain has been used for the detection of myocardial ischaemia; it may apply after coronary reperfusion to predict infarct size. It has also been suggested for patients during chemotherapy to detect a decline in cardiac function early. Similarly, strain has been proposed to estimate the risk of ventricular arrhythmias; it may apply to find the optimal position for the pacing lead in the LV free wall in the evaluation of patients after implantation of cardiac resynchronisation therapy (Smiseth et al., 2015).

Despite the vast amount of studies on strain imaging, showing its ability to detect abnormal myocardial tissue and provide a more comprehensive diagnosis, it still has some censorious drawback that is preventing its acceptance in the clinical practice. This thesis will propose a method that does not require any additional ad-hoc filtering process, which has been the major source of disagreement between the existing techniques and can potentially help to reduce the variability in the strain measurements caused by various post-processing techniques applied by different implementations of the speckle tracking.

1.3 Aims and Objectives

This PhD project forms part of a larger project which aims at developing a reliable system to automate the image analysis steps of an echocardiographic exam workflow, as illustrated in figure 1.1. This thesis has focused on automating several steps in the workflow, namely view classification, LV segmentation, and quantification of the myocardial strain. The automation of cardiac phase detection, which is an integral part of the workflow, was pursued in parallel in another PhD project, to which developments in this thesis contributed significantly.

The processing pipeline should preferably be fast enough, making it feasible for deployment in real-time applications. Nevertheless, offline analysis of already acquired images would still be immensely useful.

Therefore, the main objectives of this study are listed as below:

- Develop an automated model for view classification given two objectives of reducing the neural network size and increasing its prediction accuracy to detect various echocardiographic views.
- Develop an automated model to segment the left ventricle in detected views.
- Develop a reproducible algorithm to extract cardiac tissue movements and the resulting strain measurements, thereby allowing for the assessment of LV function.
- Evaluate the developed models/algorithms by conducting comprehensive experiments using synthetic data with known Ground-Truth (GT).

1.4 Contributions to Knowledge

Considering the novel elements of the research undertaken, the main contributions of this thesis can be summarised as follows:

View detection

- Application of the state-of-the-art neural network search technique to design efficient CNN architectures for echo view detection.
- Inclusion of 14 different anatomical echocardiographic views; larger than any previous study.
- Analysis of computational and accuracy performance of the developed models using large-scale datasets.
- Analysis of the impact of the input image resolution and size of training data on the model's performance.
- Analysis of the correlation between the image quality and accuracy of the model for view detection.

Left ventricle segmentation

- Propose a neural network model using NAS algorithmic solution to segment the LV aiming to present a model with high performance including comparison with the state-of-the-art models.
- Analysis of the performance of the developed models using two private datasets and one public dataset (Leclerc, Smistad, Pedrosa et al., 2019).
- Analysis of the impact of the size of training data on the model's performance.

Myocardial strain imaging

- Development of novel speckle tracking algorithms to extract myocardial displacements from the cardiac image sequences.
- Evaluate the fidelity of the developed algorithms comprehensively and using synthetic data for which the exact solutions are known.

During the course of this PhD study, outcomes of the research have been published in Three journal and presented at several relevant national and international conferences. In addition, one journal papers is under the fine-tuning stage to be submitted:

Journal articles:

- Azarmehr, N., Ye, X., Howes, J.D., Docking, B., Howard, J.P., Francis, D.P. and Zolgharni, M., 2020. An optimisation-based iterative approach for speckle tracking echocardiography. Medical & Biological Engineering & Computing, pp.1-15.
- Azarmehr N, Ye X, Howard P, Lane E, Labs R, Shun-shin M, Cole G, Bidaut L, Francis D, and Zolgharni M, 2020. Neural Architecture Search of Echocardiography View Classifiers, Journal of Medical Imaging
- Lane E S., Azarmehr N., Jevsikov, J., Howard J. P., Shun-shin M. J., Cole G D., Francis D.P., and Zolgharni M., 2021. Multibeat Echocardiographic Phase Detection Using Deep Neural Networks, Computers in Biology and Medicine
- Segmentation paper, to be submitted

Conference proceedings:

- Azarmehr, N., Ye, X., Janan, F., Howard, J.P., Francis, D.P. and Zolgharni, M., 2019, April. Automated Segmentation of Left Ventricle in 2D echocardiography using deep learning. In International Conference on Medical Imaging with Deep Learning
- Azarmehr, N., Ye, X., Sacchi, S., Howard, J.P., Francis, D.P. and Zolgharni, M., 2019, July. Segmentation of Left Ventricle in 2D echocardiography using deep learning. In Annual Conference on Medical Image Understanding and Analysis (pp. 497-504). Springer, Cham.
- Labs, R. B., Vrettos, A., Azarmehr, N., Howard, J.P., Shun-shin, M. J., Francis, D.P. and Zolgharni, M., 2019, July. Automated Assessment of Image Quality in 2D Echocardiography Using Deep Learning In ICRMIRO 2020: International Conference on Radiology, Medical Imaging and Radiation Oncology.
- Lane, E.S., Azarmehr, N., Jevsikov, J., Howard, J.P., Shun-shin, M., Francis, D.P. and Zolgharni, M., 2021. Echocardiographic Phase Detection Using Neural Networks.In International Conference on Medical Imaging with Deep Learning

1.5 Thesis Structure

This thesis comprises eight chapters and two appendices. In the first section of every chapter, there is an introduction on the subject of that chapter, followed by the main body:

Chapter 2 describes the clinical background of the cardiac structure, significance of echocardiography, different types of ultrasound images, different views of cardiac. Moreover, the concept of speckle tracking and corresponding challenges, and myocardial deformation parameters will be discussed.

Chapter 3 presents the technical overview of common neural networks classification and a general overview of the common CNN segmentation models. Moreover, the technical literature review of NAS solutions will be explained.

Chapter 4 investigates cardiac view classification using different deep learning methodologies. A lightweight and optimal neural network architecture have been proposed using the recent NAS solution to classify 14 echocardiographic views using a large private patient dataset of echo images. Also, the efficiency of three different state-of-the-art deep learning models for the classification of several views will be compared with the Differentiable Architecture Search (DARTS) model.

Chapter 5 explores the segmentation of LV in Two Dimensional (2D) ultrasound images using different state-of-the-art deep learning models. A neural network architecture has been proposed using the recent NAS technique to segment the LV in ultrasound images. The literature review on the segmentation of LV will be discussed.

Chapter 6 presents the related work in the STE and the feasibility of speckle tracking in the cardiac synthetic ultrasound dataset using a proposed optimised Block Matching (BM) model will be investigated. Also, the results of the proposed BM model will be compared with the standard BM model. Next, Chapter 7 will provide the experiments of the public synthetic dataset to measure strain.

Finally, Chapter 8 summarises the thesis and provides a conclusion, and presents

future directions. Finally, a list of appendices, publications, and the list of references will be presented.

1.6 Research Consortium

This study forms part of a larger 3-year British Heart Foundation (BHF)-funded collaborative project, focused on developing automated echocardiographic image analysis pipelines, involving academic/clinical partners at University of Lincoln, University of West London, Imperial College London, and St Mary's Hospital. This research is devoted to the development of systems to carry out 3 constituent steps of the overall echocardiographic examination workflow (view classification, LV segmentation, and strain measurements), while the work by other researchers involved developing deep learning models for phase detection, image quality assessment, Doppler image analysis, and electrocardiogram signal analysis.

Chapter 2

Clinical Background

2.1 Introduction

Medical imaging for cardiac function analysis is an established approach to help the clinicians to diagnose the disease and its interventional treatment. Over the past decade, the application of medical imaging techniques has facilitated state-of-the-art image-based analysis of cardiac function.

To examine the cardiac condition, there are a series of tests to measure the hearts function such as Electrocardiogram (ECG), Magnetic Resonance Imaging (MRI), Cardiac Computed Tomography (Cardiac CT), Echocardiogram (Ultrasound), etc. The echocardiogram is superior in comparison with other diagnostic imaging modalities in terms of its lower cost, higher temporal resolution, lack of X-ray exposure, and greater portability. Also, it is simple to use and able to produce a real-time moving image that is suitable for dynamic testing. However, ultrasound images suffer from lower quality (Myronenko, Song and Sahn, 2007).

In this chapter, first a brief background on cardiology will be provided, followed by a description of echocardiograms and its various modalities. Then, the speckle concept in ultrasound images, speckle tracking fundamentals and its challenges will be explained. Finally, a brief overview of the different datasets used in this thesis will be provided.

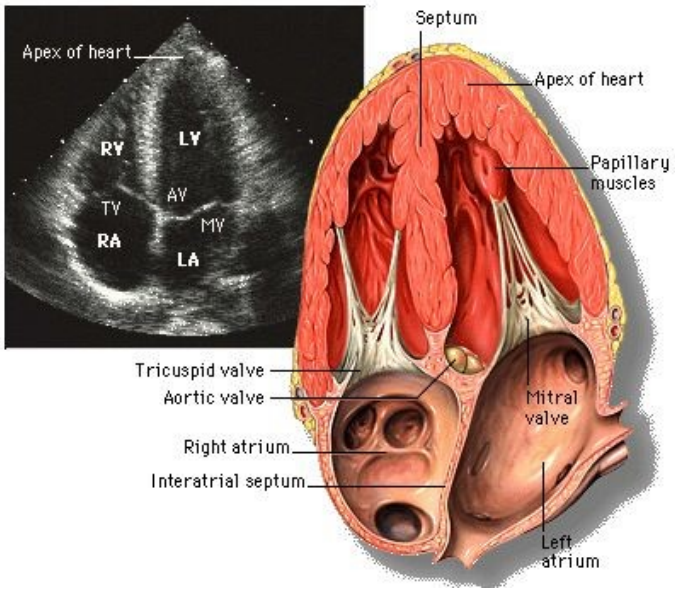


Figure 2.1: An apical four-chamber view of the heart (Yale Atlas of Echocardiology).

2.2 Overview of Cardiology

Heart is composed of three layers of tissue; a protective layer mainly made up of connective tissue, the muscles of the heart, and the inner lining of the heart which protects the valves and chambers that are called the epicardium, myocardium, and endocardium respectively. The heart cavity has four chambers, these constituting the right and left atrium above, and right and left ventricles below. The wall separating the right and left blood chambers of the heart is called septum (R. B. Hinton and Yutzey, 2011).

The heart also has one-way valves that separate the chambers and the major arteries, which prevent the back-flow of blood. Figure 2.1 depicts an Apical Four-Chamber (A4C) view of the heart; the only view that reveals all four chambers of the heart (i.e., left ventricle, right ventricle, left atrium, and right atrium). Cardiac muscle works constantly and automatically, with the LV contracting to generate pressure (a process known as systole) to squeeze blood out of the heart. Then, it relaxes to fill the heart with new blood (a process known as diastole). These sequences are known as a cardiac cycle, which forms a single heartbeat. Figure 2.2 displays phases of cardiac cycle. Being influenced by different factors such as exercise, emotions, fever, diseases, and some medication heart contract at different rates. The heart typically

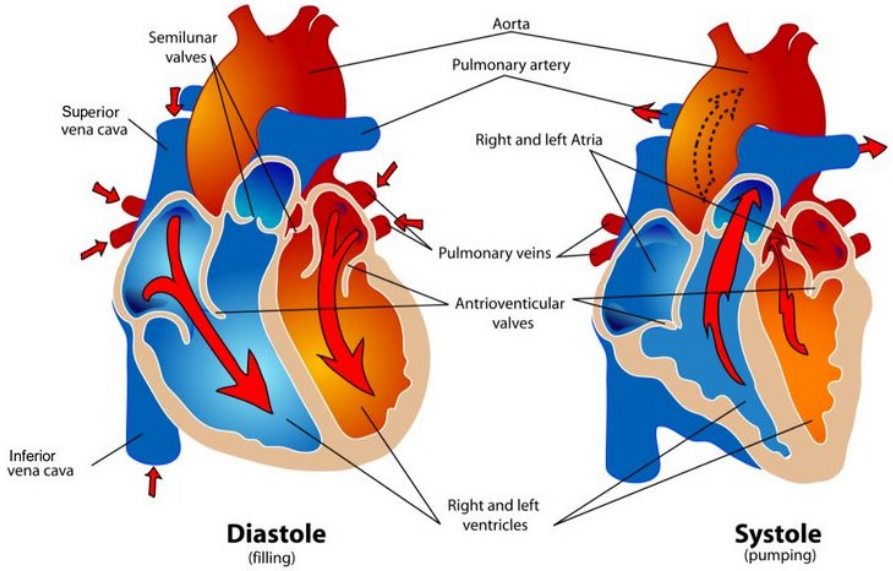


Figure 2.2: Heart shown during phases of Cardiac Cycle, periods of contraction and pumping are systole and periods of relaxation and filling are diastole (Mariana Ruiz Villarreal, 2006).

beats at around 75 beats per minute, so the length of each cardiac cycle is usually less than one second. But during this short time, a lot of pressure changes take place in the heart (Klabunde, 2011).

2.3 Significance of Echocardiograms

An echocardiogram is a non-invasive examination that uses sound waves to look at the size, shape, motion, performance of heart and its valves, pumping capacity, and the location and extent of any tissue damage. This procedure is also known as echocardiography (cardiac echo) or diagnostic cardiac ultrasound. Echocardiography can allow extracting other measures such as measuring the EF, cardiac output, and diastolic function (i.e., how well the heart relaxes) (Cleve and McCulloch, 2018).

In a patient with a suspected cardiac disorder, echocardiography is essential in assessing wall motion and to detect some cardiomyopathies diseases such as hypertrophic cardiomyopathy, dilated cardiomyopathy, etc. Also, it is helpful for early diagnosis of myocardial infarction exposing regional wall motion abnormality of the heart, in the treatment and follow-up in patients with heart failure by evaluating the EF. Echocardiography should be performed by cardiologists or sonographers trained in

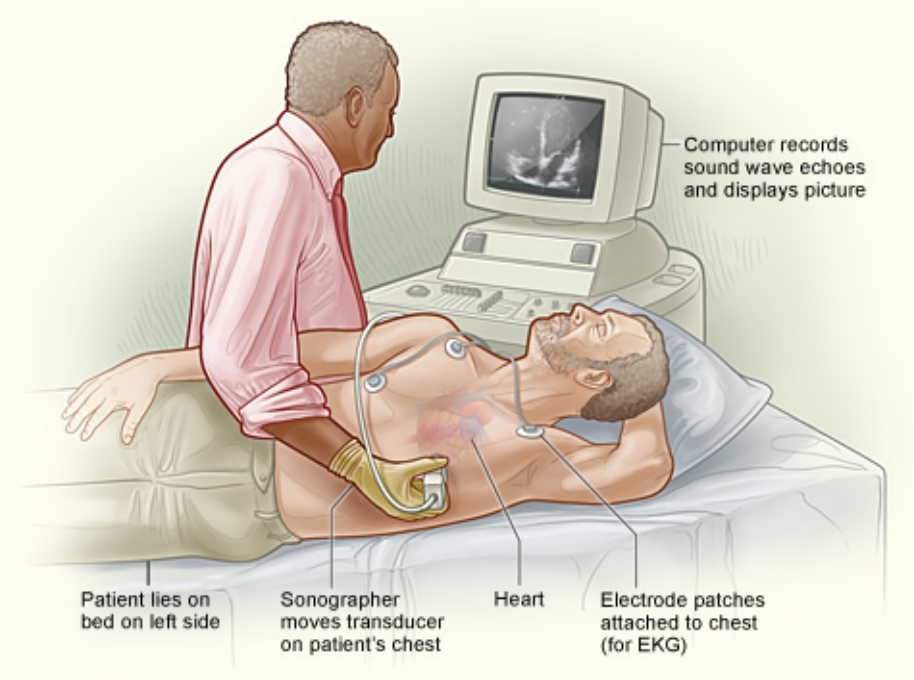


Figure 2.3: The process of Transthoracic Echocardiography (TTE) examination using the probe and viewing the cardiac on TV monitor.

echocardiography (i.e., echocardiographers) (Modin, Andersen and Biering-Sørensen, 2018).

2.4 Different Types of Echocardiograms

There exist different types of echocardiograms which can be used depending on the potential heart problem that doctors need to investigate. In the following, each type is briefly introduced.

- Transthoracic Echocardiogram (TTE):

  During this examination, a trained operator spreads a gel onto the chest and presses a device called a transducer (probe) against the skin. The transducer sends out high-frequency sound waves into the chest. This ultrasound wave will bounce off the walls and valves of the heart. The sound waves known as echoes return to the transducer and records the sound wave echoes from the heart and displays a moving image of the heart's chambers, walls, and valves on

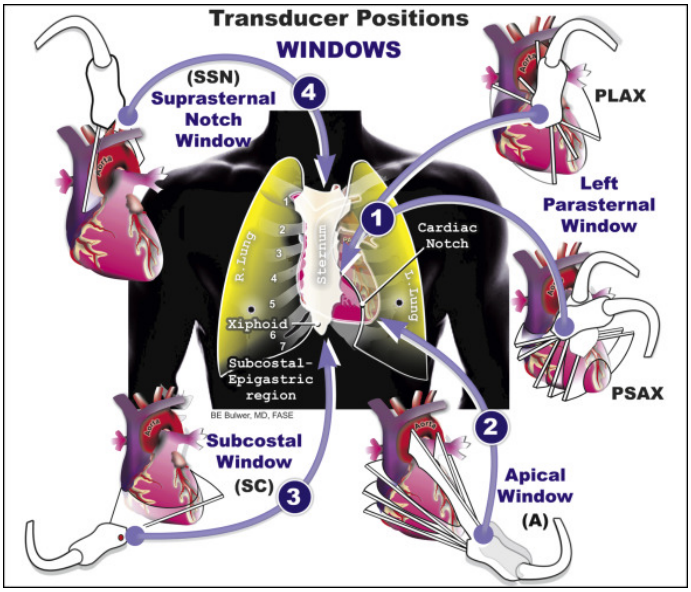


Figure 2.4: The standard recommended transducer positions in transthoracic echocardiography. PLAX: Parasternal Long-Axis, PSAX: Parasternal Short-Axis, A: Apical, SC: Subcostal, SSN: Suprasternal Notch (Bulwer, Shernan and J. Thomas, 2011).

a monitor while the scan is accomplished. The procedure of echocardiography displays in Fig 2.3.

Echocardiographic views are identified by referring to the transducer location and the imaging plane. The transducer can be placed in different locations of the chest, and at different angles, to capture the image of the heart, with these areas called 'windows'. The most common echo windows are the parasternal, apical, subcostal, and suprasternal. Fig 2.4 displays the most common echo windows.

From each window, the transducer can be manipulated to obtain multiple views of the heart. The different views can be attained by rotating and/or tilting the transducer without moving it to a new window. The quantification of the cardiac function and chambers with echocardiography has appeared as a dominant technique in the detection and assessment of cardiac disease because of its exclusive ability to provide real-time images of the beating heart (Horton, 2010). This thesis has focused on TTE as the most frequent type of echocardiography used in clinical practice.

- Stress Echocardiogram:

  Stress echocardiogram examination known as stress echo is to find out if the patient has decreased blood flow to the heart muscle (i.e., coronary artery disease). The stress echo uses ultrasound imaging of the heart to evaluate the wall motion in response to physical stress. This examination increases the heart rate and blood pressure. During this examination, two sets of images will take including one at rest, and another after working out on a treadmill or stationary bike. If the patient health condition limits physical activity, a medication will be injected to simulates the effect of exercise (Prisant, Watkins and Carr, 1984).

- Transesophageal Echocardiogram (TEE):

  For TEE examination, the transducer instead of being moved over the outside of the chest wall is passed down the oesophagus. TEE would allow producing clearer pictures of the heart because the transducer is located closer to the heart and the lungs and bones of the chest wall do not block the sound waves generated by the transducer (O'Rourke and Mendenhall, 2019; Blinn, Margulis and Joshi, 2019).

- Three Dimensional Echocardiogram:

  A Three Dimensional (3D) echocardiogram uses either transesophageal or transthoracic echocardiography to generate a 3D image of the heart. This examination includes multiple images from various angles. It's used for recognising problems with heart valves, before heart valve surgery for replacement heart valves, or diagnosis heart problems in children (Lang, Mor-Avi et al., 2006). Currently, 3D echocardiography suffers from a considerable reduction in frame rate and image quality, and this has hindered its adoption into routine practice. When such issues are resolved, automatic analysis of the 3D images could also be explored. Meanwhile, 2D echocardiography remains unrivalled and clinically relevant, particularly when high frame rates are needed.

2.5 Different TTE Modalities

A-Mode (Amplitude Mode):

A-mode is an operational situation that a system has been transferred to, and a normal mode occurs when all sections of a system oscillate with the same rate of occurrence. For ultrasound imaging of the heart, diverse types of mode can be controlled by the operator, each of them conveying a specific type of information to the clinician. There are three basic modes used to image the heart which is briefly explained in the following section:

- M-Mode (Motion Mode):

  This mode is used to reflect the motion of an organ. Its application is also used in cardiac timing and the measurement of dimensions. M-Mode displays a one-dimensional image that measures the distance of the object from the single transducer at a given moment. The ultrasound shows this information as a 2D image which is depth and time. M-mode images have very high sampling rate, which results in a high time resolution. Therefore, very rapid motions can be recorded, displayed, and measured. However, in these type of images, the ultrasound line is fixed to the tip of the ultrasound sector. It may therefore be difficult to align the M-mode perpendicular to the structures which are displayed (i.e. the septum), thus leading to false measurements (Loizou, Pattichis and D'hooge, 2018).

- B-Mode (Brightness Mode):

  This mode is more commonly known as 2D that allows a plane of tissue (both depth and width) to be imaged and displays the ultrasound reflection as an 8bit greyscale image that composed of bright dots representing the ultrasound echoes. The brightness of each dot is determined by the amplitude of the returned echo signal. This allows for visualization and quantification of anatomical structures, as well as for the visualization of diagnostic and therapeutic procedures. The anatomic relationship between various structures is easier to recognise than M-mode echocardiographic images.

The formation of a B-mode image depends on the pulse-echo principle; assuming the speed of sound remains constant, the position of a target of interest may be inferred by the time taken from emission to its return to the transducer. The limitless number of imaging plane through the heart is possible, however, the standard view will use to assess the intra and extra cardiac structure. This thesis focuses on 2D imaging (Prada et al., 2015).

- TDI (Tissue Doppler Imaging):

  Tissue Doppler Imaging (TDI) is a modality that allows to measure myocardial velocities to evaluate global and regional myocardial systolic and diastolic function. It can also be employed to quantify right ventricular and left atrial function (Atzeni et al., 2017). TDI is useful as a diagnostic as well as prognostic tool in different cardiac conditions such as coronary artery disease, heart failure (both systolic and diastolic), valvular heart disease, cardiomyopathies and constrictive pericarditis. Also, TDI measurements are helpful to recognise patients who will benefit from cardiac resynchronisation therapy. Although TDI is reproducible and quite easy to acquire, it is underutilised in routine clinical practice. (Kadappu and L. Thomas, 2015).

  Fig 2.5 displays some examples of different modalities used in transthoracic echocardiography.

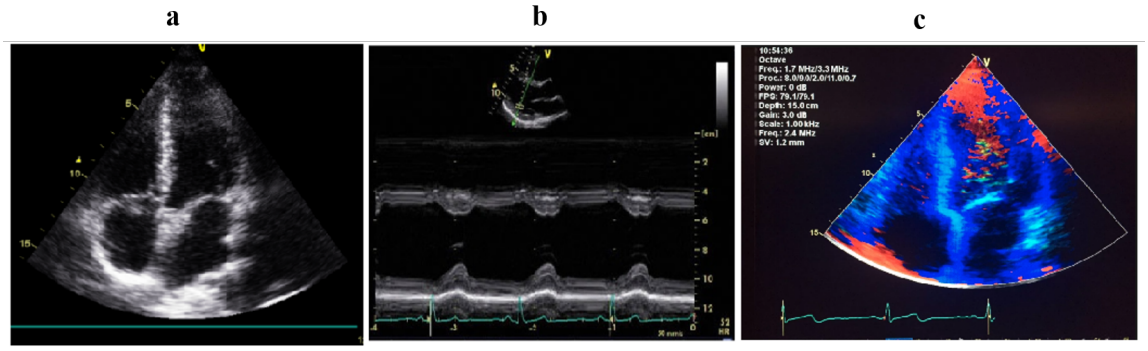


Figure 2.5: Examples of different modalities used in transthoracic echocardiography. a: B-Mode (2D imaging), b: M-Mode, c: TDI

2.6 Speckle Tracking

The granular appearance of an image produced by the interference of ultrasound waves in the tissue is known as 'speckle'. This occurs when a random group of scatterers is illuminated by waves bearing a wavelength larger than the size of the individual scatterers, which causes speckles to appear.

Since in medical ultrasound imaging, soft tissues comprise of many scatterers, the ultrasound waveforms perceived by the transducer are a group of diverse wave reflections caused by the distinct scatterers, with generated speckles observable in the unfiltered grey-level (2D imaging) images where they appear as dark, bright specks.

These speckles can be used to provide predictions of some cardiac events and help with the primary diagnosis of myocardial changes. Since speckles are identifiable by the conventional 2D greyscale echocardiography, and speckle tracking is independent of the angle of insonation, assessment of the cardiac mechanism is conceivable in three-spatial orientations, such as longitudinal, radial, and circumferential (Abduch et al., 2014).

These speckles exhibit a pattern that is assumed to be unique for each myocardial segment that remains approximately stable under the same acquisitions during the cardiac cycle (from frame to frame). Tracking these speckles, and analysing them frame by frame can help to quantify the myocardial function and will allow the extraction of some parameters such as displacement and strain that will be explained in section 2.7 (Bansal and Kasliwal, 2013).

STE is a relatively novel imaging modality that was first presented by (Reisner et al., 2004). It has developed quickly from a research tool to a technology tool that sits on the threshold of becoming part of routine echocardiography (Blessberger and Binder, 2010). STE is one of the most outstanding and non-invasive ultrasound imaging methods used to acquire quantifiable information regarding myocardial deformation, motion, and function evaluation (Curiale, Vegas-Sánchez-Ferrero and Aja-Fernández, 2016).

2.7 Myocardial Deformation Parameters

In this section, the concept and the significance of myocardial measures such as displacement, velocity and strain that can be obtained by speckle tracking during an assessment of cardiac function will be discussed.

2.7.1 Displacement and Velocity

Displacement defines the distance that certain speckle features (cardiac structure) have moved between two consecutive frames. Velocities also reproduce displacement per unit of time, that accounting for how fast the location of a speckle feature changes. Since velocity and displacement are vectors, they have direction and amplitude. Therefore, they can be examined through different spatial movements along with the anatomic coordinates of the cardiac chambers, longitudinal, radial and circumferential components, which are especially relevant for the characterization of myocardial mechanics (Mor-Avi et al., 2011).

2.7.2 Global and Regional Strain

Over time, a moving object will change its position (displacement), but will not deform if all its parts move with the same velocity. On the other hand, if different parts of the object move with different velocities, the object will change its shape and go through deformation. Therefore, the displacement and velocity of wall motion are not enough to distinguish between the active and passive movement of myocardial segments. However, strain can differentiate between active and passive myocardial tissue movement (Dandel, Lehmkuhl et al., 2009).

Strain is a significant parameter in the quantification of myocardial function. Strain describes the change of myocardial tissue length when compared to its original length. Fundamentally, strain measures the extent (intensity) of the contraction and relaxation of myocardial tissue. When myocardial tissue is thinning or shortening strain is negative, whereas when it is lengthening or thickening strain gives a positive value. It is defined mathematically as the change of myocardial tissue length during stress

at the end-systole which is compared with the original length in a relaxed state at the end-diastole (Pavlopoulos and Nihoyannopoulos, 2008).

During the cardiac cycle, as the LV contracts, the muscle shortens in the longitudinal and circumferential dimensions which can produce a negative strain. Also, the muscle will thicken or lengthen in the radial direction to produce a positive strain (Thomas H Marwick, 2006). The application of strain to measure deformation is one of the objectives that this thesis intends to investigate, as it can be used to indicate the health of a patient's heart. To examine the strain on the heart's muscle imaging of the heart is required, and this can be achieved in numerous ways. This study focuses on echocardiography, or imaging of the heart using an ultrasound scanner.

Strain has been used for the detection of myocardial ischaemia; it may apply after coronary reperfusion to predict infarct size; it is suggested for patients during chemotherapy to detect a decline in cardiac function early. Similarly, the strain has been proposed to estimate the risk of ventricular arrhythmias; it may apply to find the optimal position for the pacing lead in the LV free wall in the evaluation of patients after implantation of cardiac resynchronisation therapy (Smiseth et al., 2015).

A more detailed discussion to strain measurements is provided at the beginning of Chapter 7.

2.8 Overview of Datasets Used in the Thesis

Representative multi-centre patient datasets are essential for ensuring that any developed models would scale up well to other sites and environments. Therefore, this study employed several private and public datasets, originating from different clinical sites and acquired by different imaging equipment from various vendors, and representative of real-world patient population. Table 2.1 provides a brief summary of the datasets used. Chapter 4, 5 and 6 will provide the details of each dataset, when dealing with problem-specific data.

Table 2.1: Summary of echocardiographic patient datasets used for different applications/tasks including classification, segmentation and Speckle Tracking in this project.

Dataset Name	PACS	CAMUS	EchoLab	Synthetic
Use	Classification, Segmentation	Segmentation	Segmentation	Speckle Tracking
Type	Private	Public	Private	Public
Source	NHS Trust, Imperial College Healthcare	University Hospital of St Etienne (France)	NHS Trust, Imperial College Healthcare	Alessandrini et al.,2017
Ultrasound machine	GE and Philips	GE Vivid E95	Philips iE33	-
No of Patients	374	450	61	-
No of Videos	8732	450	61	14
Ground-truth	1 annotations	1 annotations	2 annotation by 2 experts	1 annotations
No of Frames	41321	1800	992	Different for each vendor

2.9 Conclusion

This chapter provided the clinical background of cardiology, the significance of echocardiograms, the different types of echocardiograms and ultrasound modalities with a focus on the Transthoracic Echocardiography (TTE) and B-Mode modality where our dataset is gathered from this tool and modality. Moreover, the speckle tracking concept and myocardial deformation parameters have been discussed.

Chapter 3

Technical Background

Recently, advances in AI using deep learning has been used as a developing tool to assist diagnosis in the medical domain (LeCun, Bengio and G. Hinton, 2015; Shen, G. Wu and Suk, 2017; SUZUKI, 2017). Also, AI may have potential in the evaluation, diagnosis and prognosis of cardiovascular disease (Shrestha and Sengupta, 2018; Tabassian et al., 2018).

In contrast with classical machine learning that concerns the derivation of predefined features in the input image, deep learning allows predicting the results automatically without pre-defined imaging features (M. I. Jordan and T. M. Mitchell, 2015). Furthermore, CNN enable to extract high and low-level information from the input image and combine these to create higher-order structural information, enabling the identification of the complicated structures from the images (LeCun, Bengio and G. Hinton, 2015; Amari et al., 2003).

In this chapter, first, an overview of neural networks will be presented, including a brief introduction to CNN and different approaches to neural network design. Then, the general CNN architectures used for classification and segmentation will be described. Lastly, an overview of NAS will be illustrated.

3.1 Overview of Neural Networks

Deep learning is a subset of machine learning in AI that has become the "crown jewel" of AI (LeCun, Bengio and G. Hinton, 2015). Deep learning methods have emerged as robust techniques for learning feature representation automatically from

the data (G. E. Hinton and Salakhutdinov, 2006). Also, Deep learning methods have provided major improvements in the classification and segmentation of medical images (Litjens et al., 2017; Ronneberger, Fischer and Brox, 2015; Shen, G. Wu and Suk, 2017). Deep learning methods work toward learning feature hierarchies, where features at higher levels of the hierarchy are formed using the features at lower levels (Dean, 2016). In the following, CNNs are introduced, thereby providing a foundation for describing the techniques used in this project.

3.2 Introduction to CNN

Convolutional neural networks are a kind of feed-forward artificial neural networks that the connectivity pattern between its neurons is inspired by the human brain. CNNs with a special multi-layer neural network will train with a version of the back-propagation algorithm. CNNs are designed to recognise the visual patterns directly from pixel images. CNNs are also composed of layers such as input layer, convolutional layer, activation layer, pooling layer, and Fully Connected layer (FC) (Kalchbrenner, Grefenstette and Blunsom, 2014):

- Input Layer: The input layer holds the raw pixel values of the input data that in the case of the echocardiographic data; the width and height of the input layer are the spatial dimensions of a single frame which will be exposed to the network.

- Convolutional Layer: The convolution layer extracts features from the input image using the kernels/filters (i.e. fixed size matrices). Kernels are sliding (convolving), across the input image and the element-wise multiplication will be computed between the values in the kernel matrix and the original image value. The output of these operations will be called feature map. An example of convolution operation is illustrated in Figure 3.1. The more kernels we have, the more image features will get extracted and the network become better at recognising the pattern in unseen images (test dataset).

  The size of the feature map will be controlled by the number of kernels (depth),

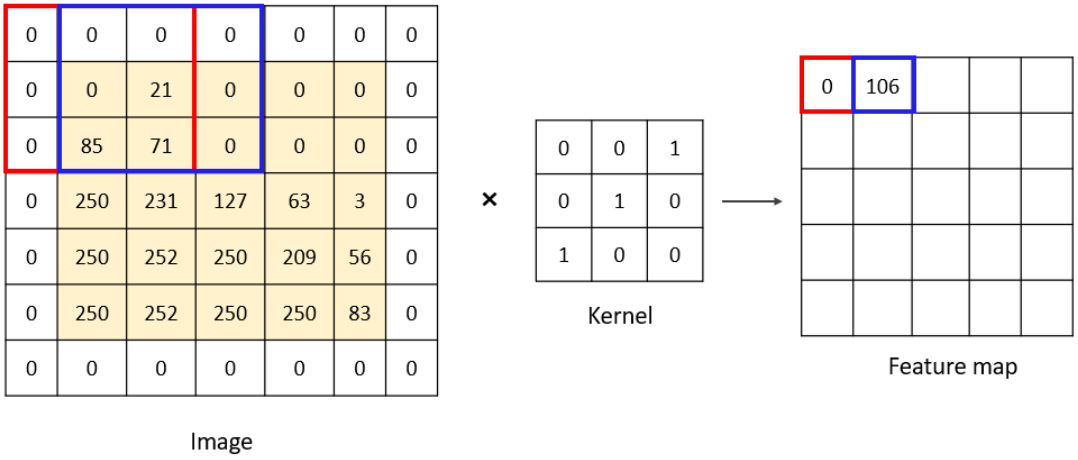


Figure 3.1: Illustration example of convolution operation

the number of pixels that slide over the input image (stride), and padding the input image with 0s around the border (zero-padding).

- Activation Layer: The activation layer is a non-linear function to convert the output of the convolutional layer to an output that can be used in the next layer. In the past, non-linear functions like tanh and sigmoid were used, however, the Rectified Linear Unit (ReLU) layer works far better because the network is also able to train a lot faster without making a significant difference to the accuracy.

  The ReLU layer applies the function f(x) = max(0, x) to all of the values in the input i.e if x < 0 , f(x) = 0 and if x >= 0 , f(x) = x . Therefore, this layer will change all the negative activations to 0, and this leaves the size of the feature map unchanged. The ReLU visually looks like the Figure 3.2. This layer increases the nonlinear properties of the model, and in the literature, the ReLU is the most activation function used in deep learning methods as it can learn fast in the large neural networks (G. E. Hinton, 2010). The networks trained with the RelU function almost prevent the problem of vanishing gradient, allowing models to learn faster and perform better. The vanishing gradient term refers to the fact that in a neural network the backpropagated error usually increases

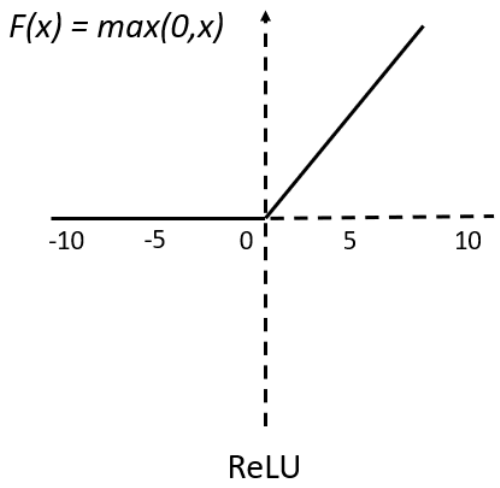


Figure 3.2: ReLU Activation Function

or decreases exponentially as a function of the distance from the final layer (Sussillo and Abbott, 2015).

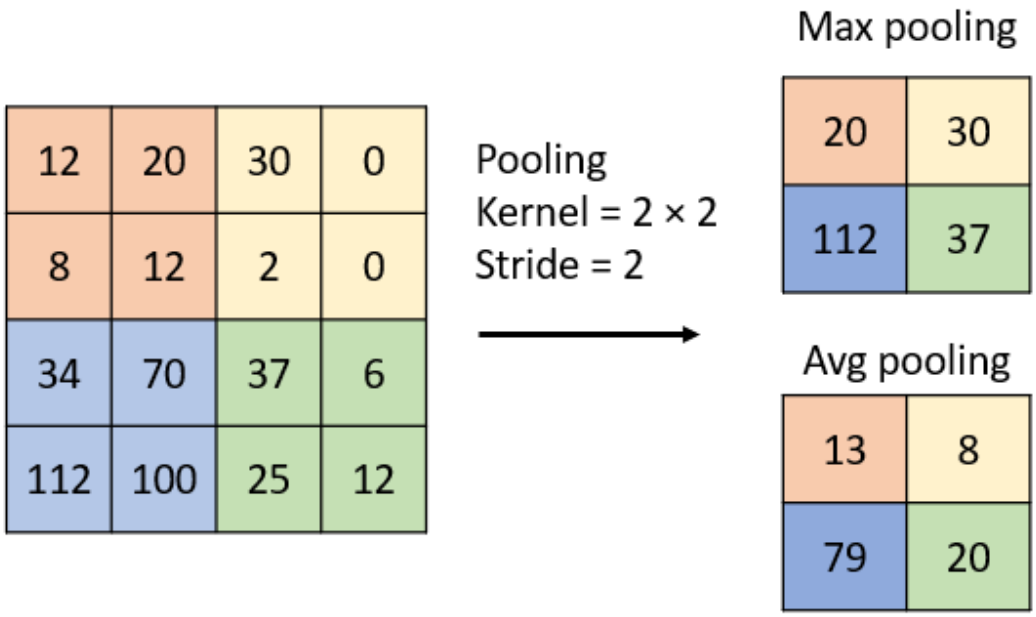


Figure 3.3: Illustration example of max pooling and average pooling with the kernel size of 2×2 and stride of 2. Digits express of the max and average pooling operations.

- Batch Normalisation: the batch normalisation operation introduced by (Ioffe and Szegedy, 2015) is used to standardise each layer's input to have zero mean and unit variance. Generally, batch normalization aims to make the distribution of inputs to a given network layer more steady during the training. This

is can be accomplished by augmenting the network with additional layers that set the mean and variance of the distribution of each activation to be zero and one respectively. Then, to preserve model expressivity, the batch normalization inputs need to be scaled and shifted based on the trainable parameters. This normalization will be applied before the non-linearity of the previous layer. Batch normalization allows faster and more stable training of deep neural networks (Santurkar et al., 2018).

- Dropout: dropout is a technique that is introduced by (G. E. Hinton, Srivastava et al., 2012). During the training phase, random samples of neurons will be ignored and these neurons will not be considered during a particular forward or backward pass. The main purpose behind the dropout technique is to prevent overfitting. Dropout intended to change the network architecture randomly to reduce the risks that the learned weight values get very adopted to the underlying training data, and consequently cannot be generalised well to test data (Garbin, X. Zhu and Marques, 2020).

- Pooling Layer: The pooling layer are responsible to reduce progressively the spatial size of each slice of feature map independently using the maximum or average operation and reduce the number of parameters and computations in the network to control the overfitting model. The size of the pooling operation or filter is smaller than the size of the feature map.

  For example, if we have 4×4 matrix representing our initial input, and a 2×2 kernel that will run over the input with the stride of 2, the maximum or the average of that region will take and create a new output matrix where each element is the max of a region in the original input. An illustration example displays in Figure 3.3.

- Fully Connected Layer: To wrap up the CNN architecture the fully connected layer will place after the convolution and pooling layer. The output of the convolution and pooling layers are represented of high-level features of the input image, and the FC layer uses these features for classifying the input image into different classes based on the training dataset. Neurons in a fully

connected layer have a full connection to all the activations in the previous layer.

3.3 Approaches to Neural Network Design

Neural networks are extremely effective at learning patterns and features from digital images and have demonstrated success in many image classification and semantic segmentation applications over the past few years (Krizhevsky, Sutskever and G. E. Hinton, 2012; Litjens et al., 2017). However, the manual process of designing the architecture has been accompanied by a growing demand for architecture engineering of increasingly more complex deep neural networks through a time-consuming and arduous manual process that deeply relies on expert domain knowledge.

Moreover, the developed architectures are usually dependent on the particular image dataset used in the design process, and adapting the architectures to new datasets remains a very difficult task that relies on extensive trial and error process and expert knowledge.

Inspired by the AutoML (Cai, L. Zhu and Han, 2019; H. Liu, Simonyan and Yang, 2019; Hutter, Kotthoff and Vanschoren, 2019), there is a growing interest in algorithmic solutions, such as NAS to automate the manual process of architecture design and to discover better neural network architectures with better performance, and fewer parameters, and even lower computation cost to speed up the inference for classification and semantic segmentation.

Pivotal to the NAS architecture, is the creation of a large collection of potential network architectures. These options are subsequently explored to determine an ideal output with a specific combination of training data and constraints, such as network size. Initial NAS approaches, such as Reinforcement Learning (RL) (Bello et al., 2017; Zoph, Vasudevan et al., 2018) and evolution (Real et al., 2019b), search for complete network topology, thus involving extremely large search spaces comprised of arbitrary connections and operations between neural network nodes. Such complexity results in using massive amounts of energy and requiring thousands of GPU

hours or million-dollar cloud compute bills (Strubell, Ganesh and McCallum, 2019) to design neural network architectures.

As discussed in the 3.2 section, the main layers of a CNN are used to create networks for different purposes. Different types of neural networks are constructed to learn and provide informative features for the classification and segmentation models. In the following, the most common classification and segmentation architectures that wildly have been used will be explained. This will be followed by an overview of NAS methods applied to the classification and segmentation tasks.

3.4 General Classification Architectures

As discussed in the previous section, the main layers of a CNN are used to construct networks for different purposes. In literature, several CNN architectures are created to learn and provide informative features for the classification models including LeNet (LeCun, Bottou et al., 1998), AlexNet (Krizhevsky, Sutskever and G. E. Hinton, 2012), VGG'16 (Simonyan and Zisserman, 2015), GoogleNet (Szegedy, Wei Liu et al., 2015), ResNets (He et al., 2016), DenseNet (G. Huang et al., 2017), etc. In the following, the most common CNN architectures that have been widely used will be illustrated.

- **LeNet:** The LeNet architecture proposed by (LeCun, Bottou et al., 1998) for handwritten and machine-printed character recognition in the 1990s. The LeNet neural network contains two sets of convolution, activation and pooling layers, followed by a Fully Connected (FC) layer, activation and another FC layer and finally a softmax layer.
- **AlexNet:** The AlexNet architecture include eight layers such as five convolutional layers and three FC layers. After each convolution and FC layer, ReLU is applied (Krizhevsky, Sutskever and G. E. Hinton, 2012).
- **VGGNet:** The VGGNet is a well-known convolutional neural network proposed by (Simonyan and Zisserman, 2015) and was used to win ILSVR (Large Scale Visual Recognition Challenge, 2014) competition. This model makes

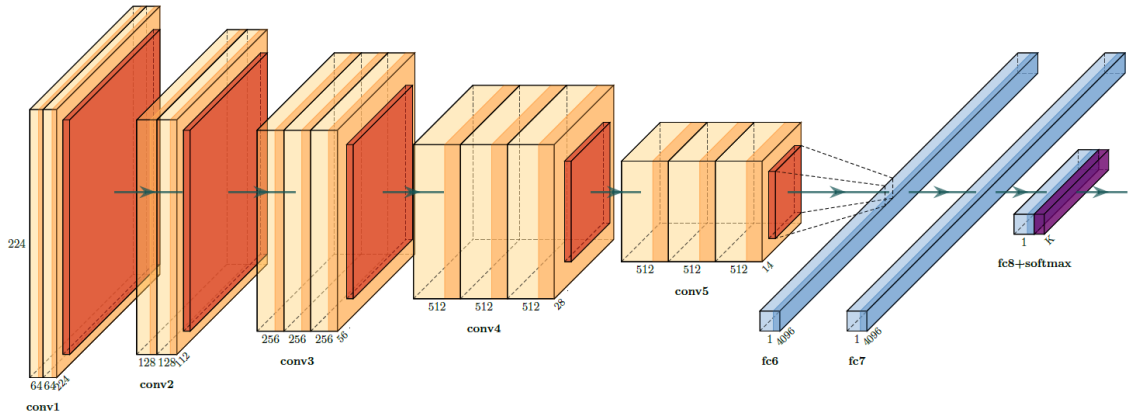


Figure 3.4: Illustration of the VGG'16 architecture for image classification proposed in (Simonyan and Zisserman, 2015)

an improvement over AlexNet architecture (Krizhevsky, Sutskever and G. E. Hinton, 2012) by increasing the depth of the network. The architecture is comprised of building blocks of two convolutional layers followed by a pooling layer. This block is repeated multiple times, whilst all the convolution kernels are of size 3×3. Finally, a stack of convolutional layers is followed by three FC layers. Also, by introducing the number of layers (i.e. 11, 16 and 19) different architectures were proposed and the VGG'16 with a total of 16 layers as shown in Figure 3.4 recommended to have the best performance.

- **GoogleNet:** The GoogLeNet neural network is introduced by (Szegedy, Wei Liu et al., 2015). They proposed an Inception Module that reduces the number of parameters in the architecture. The Inception Module applies multiple convolutional filters for the same input and concatenates the result. The network is consist of 22 layers.

- **ResNet:** Experiments have revealed if layers stacked in the network without changing the network structure, the performance of the network would get worse because gradients of network parameters will vanish as the depth is increasing. To solve this challenge, He et al. (2016) present ResNet, which suggested a residual learning framework through adding identity-mapping shortcuts (He et al., 2016). The ResNet model utilises four modules comprising residual blocks, each of which uses several refers blocks with the same number

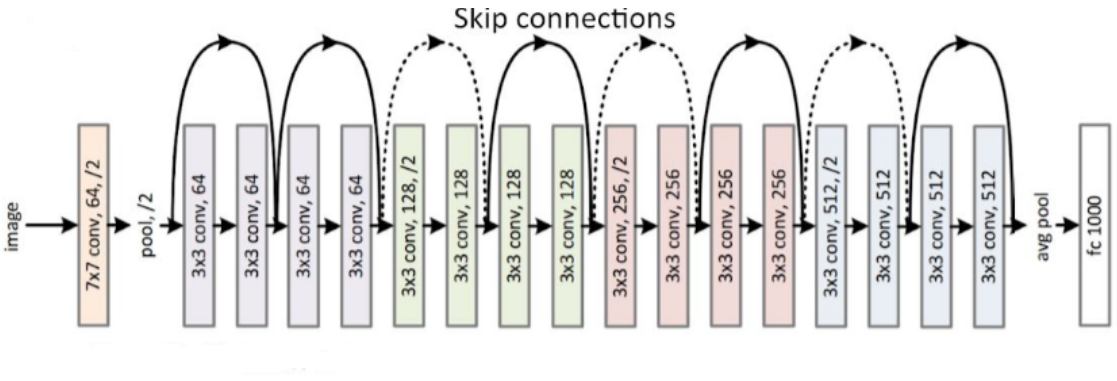


Figure 3.5: Illustration of the ResNet-18 architecture

of output channels. The number of channels in the first module is the same as the number of input channels. Each residual block has two 3×3 convolutional layers with the same number of output channels. Each convolutional layer is followed by a batch normalization layer and a ReLU activation function, except the last operation of a block, that does not have the ReLU. There are 18 layers in total. Therefore, this model is commonly known as ResNet-18. By configuring different numbers of channels and residual numbers of channels and residual blocks in the module, different ResNet models have been created such as ResNet-152 (He et al., 2016). Figure 3.5 illustrates the architecture of ResNet-18.

- **DenseNet:** The DenseNet presented by (G. Huang et al., 2017) that in a feed-forward fashion, connects each layer to every other layer. It includes a convolution operation or pooling layers, batch normalization, and an activation function. DenseNet concatenates the output feature maps of the layer with the incoming feature maps as illustrated in Figure 3.6.

The depth of the CNN network shows a significant impact on the performance of the model, however, getting deeper without applying changes in the structure can cause a poor performance, lead to loss of information and vanishing-gradient problem (Wenqi Liu and K. Zeng, 2018). To overcome these challenges, Huang et al. (2017) presented DenseNet architecture which reduces the number of parameters, enhance the gradient and information flow throughout the network that make the network easier to train. In addition, DenseNet achieves better feature reuse through connecting the output

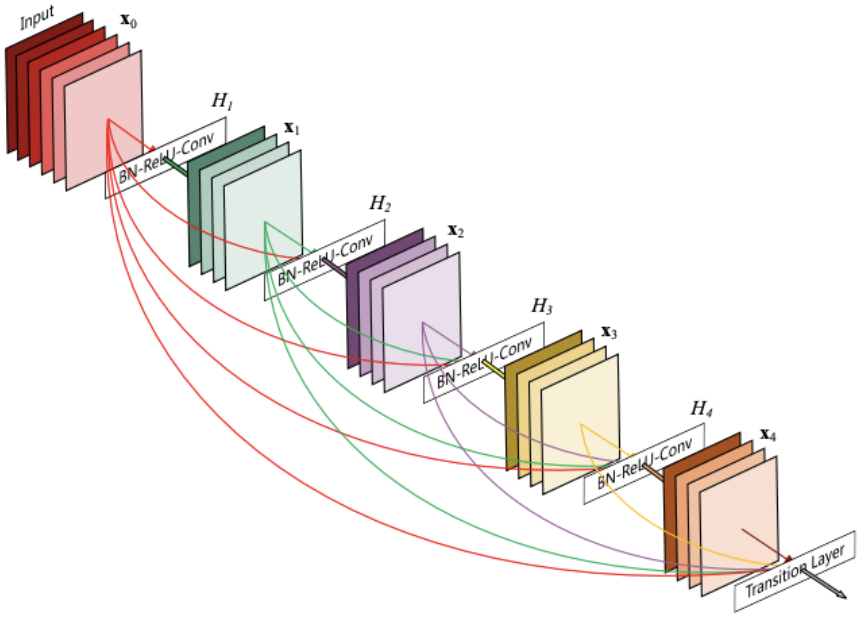


Figure 3.6: Illustration of a 5-layer dense block (G. Huang et al., 2017)

of each layer to another layer (G. Huang et al., 2017). This thesis investigates the performance of VGG16, ResNet18, and DenseNet201 classification models to classify different views in echocardiography images.

3.5 General Segmentation Architectures

With the promising capacity of a CNN in the image classification task, applying a CNN to medical image segmentation has been explored by many researchers. The general idea is to perform segmentation by using a 2D input image and applying 2D filters on it (Hesamian et al., 2019). Image semantic segmentation aims to obtain pixel classification of an image. For this goal, researchers introduced the encoder-decoder structure such as Fully Convolution Network (FCN) (Long, Shelhamer and Darrell, 2015), U-Net (Ronneberger, Fischer and Brox, 2015), Deeplab (L.-C. Chen, Papandreou, Schroff et al., 2017), etc. In these structures, an encoder is often used to extract image features while a decoder is often used to restore extracted features to the original image size and output the final segmentation results. In the following general segmentation architectures will be presented.

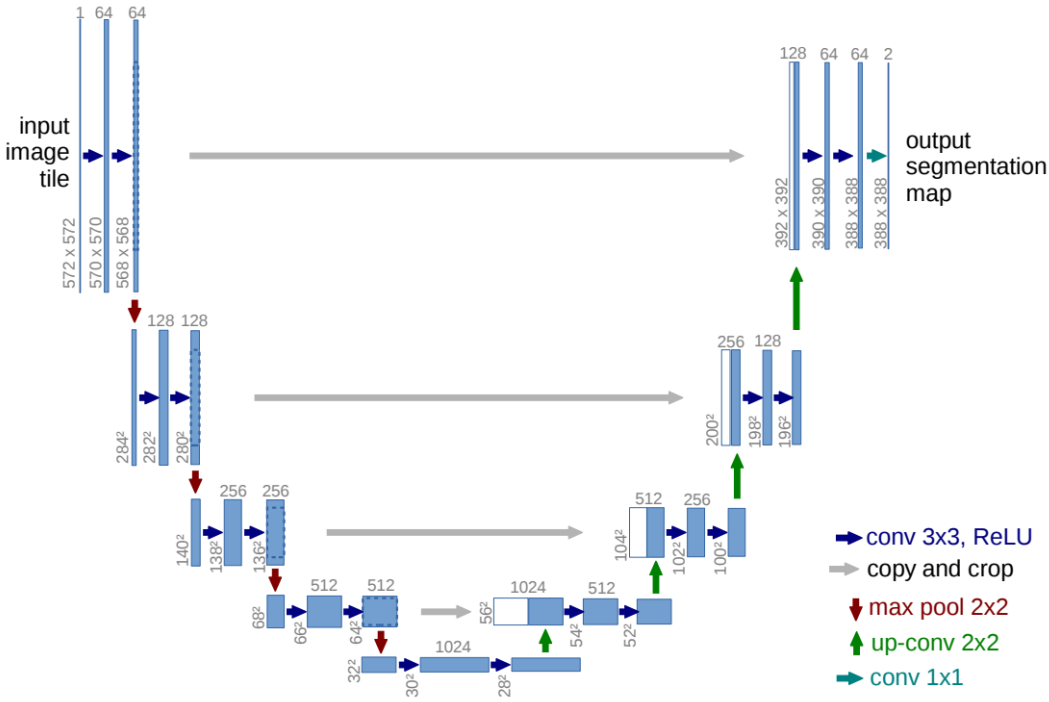


Figure 3.7: U-Net architecture (example for 32×32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations (Ronneberger, Fischer and Brox, 2015).

- **FC-DenseNet:** FC-DenseNet model is a relatively more recent model which consists of a downsampling and up-sampling path made of dense block. The down-sampling path is composed of two Transitions Down (TD) while an up-sampling path is containing two Transitions Up (TU). Before and after each dense block, there is concatenation and skip connections. The connectivity pattern in the up-sampling is different from the down-sampling path. In the down-sampling path, the input to a dense block is concatenated with its output, leading to linear growth of the number of feature maps, whereas in the up-sampling path, it is not (Jégou et al., 2017).

- **U-Net:** Standard and well-established U-Net neural network architecture has been successfully applied to multiple medical image segmentation problems (Ronneberger, Fischer and Brox, 2015). The U-Net architecture comprises of three main steps such as down-sampling, upsampling steps and cross-over con-

nections. During the down-sampling stage, the number of features will increase gradually while during up-sampling stage the original image resolution will recover. Also, cross-over connection is used by concatenating equally size feature maps from down-sampling to the up-sampling to recover features that may be lost during the down-sampling process. Each down-sampling and up-sampling has five levels, and each level has two convolutional layers with the same number of kernels ranging from 64 to 1024 from top to bottom correspondingly. All convolutions kernels have a size of (3×3). For downsampling Max pooling with size (2×2) and equal strides was used. U-Net architecture has a lot of attention in medical image segmentation and based on which many variations have been developed (Çiçek et al., 2016; Gordienko et al., 2018; G. Zeng et al., 2017).

- **SegNet:** The SegNet model contains an encoder stage, consists of 13 convolutional layers which correspond to the first 13 convolutional layers in the VGG16 network (Simonyan and Zisserman, 2015) designed for object classification. Each encoder layer has a corresponding decoder layer and hence the decoder network has 13 layers. The final decoder output is fed to a multi-class softmax classifier to produce class probabilities for each pixel independently. In SegNet model, to accomplish non-linear up-sampling, the decoder performs pooling indices computed in the max-pooling step of the corresponding encoder (Badrinarayanan, Kendall and Cipolla, 2017). The number of kernels and kernel size was the same as the U-Net model. This architecture is illustrated in Figure 3.8.

- **DeepLab:** DeepLab is another state-of-the-art semantic segmentation model introduced by Google (L.-C. Chen, Papandreou, Kokkinos et al., 2017). Multiple improvements have been made to the model since then, including DeepLab V2, DeepLab V3, and the latest DeepLab V3+. The DeepLab model is based on combining two popular neural network architectures such as spatial pyramid pooling and encoder-decoder networks. Spatial pyramid pooling networks can encode multi-scale contextual information. This is done using pooling operations at multiple rates. DeepLab has introduced the concept of

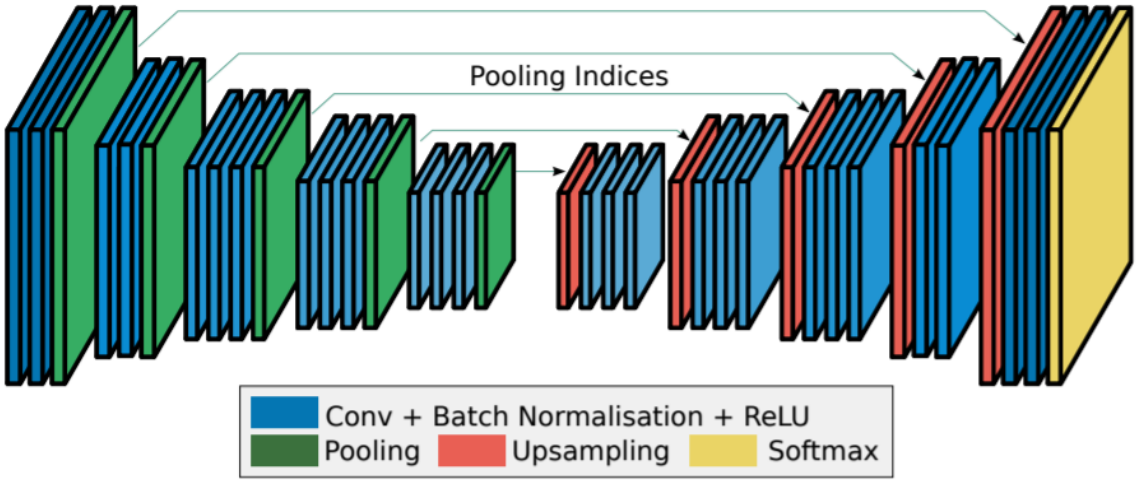


Figure 3.8: An illustration of the SegNet architecture. A decoder upsamples its input using the transferred pool indices from its encoder to produce a sparse feature map(s). It then performs convolution to densify the feature map. The final decoder output feature maps are fed to a softmax classifier for pixel-wise classification (Badrinarayanan, Kendall and Cipolla, 2017).

atrous convolutions which require a parameter called rate that would be used to explicitly control the effective field of view of the convolution. The normal convolution is a special case of atrous convolutions with a rate of 1.

The Atrous Spatial Pyramid Pooling (ASPP) introduced in deepLab model that uses spatial pyramid pooling with atrous convolutions. ASPP uses atrous convolution with rates 6, 12, and 18. It also adds image-level features with global average pooling. Bilinear upsampling also is used to scale the features to the correct dimensions. Deeplabv3-ResNet101 is constructed by a Deeplab v3 model with a ResNet-101 backbone using atrous convolution (He et al., 2016).

- **UNet++ (Nested U-Net):** UNet++ introduced an architecture for medical image segmentation. This network starts with an encoder sub-network or backbone followed by a decoder sub-network. There are re-designed skip pathways (green and blue) that connect the two sub-networks and the use of deep supervision (red) as shown in Figure 3.9 (Z. Zhou et al., 2018).

  This thesis investigate the performances of U-Net, SegNet, Deeplabv3-ResNet101,

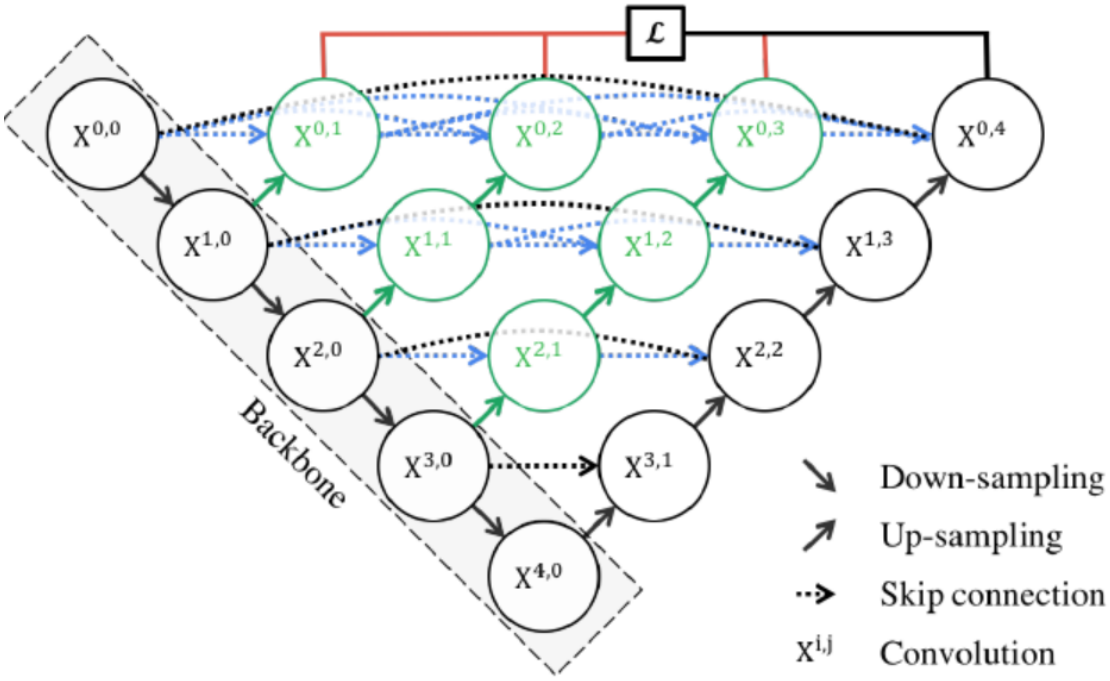


Figure 3.9: UNet++ architecture consists of an encoder and decoder that are connected through a series of nested dense convolutional blocks. The black indicates the original U-Net, green and blue show dense convolution blocks on the skip pathways, and red indicates deep supervision.(Z. Zhou et al., 2018).

and UNet++ dense prediction models to segment LV in echocardiography images.

3.6 Overview of the Neural Architectures Search

Recently, several related algorithms for NAS have arisen. NAS solutions aims to create a network architecture with the best performance automatically with less human intervention. Pioneering work of NAS are NAS-RL (Zoph and Le, 2017), MetaQNN (Baker et al., 2017) and RL methods.

NAS techniques have outperformed manually designed architectures on some tasks such as image classification (Zoph, Vasudevan et al., 2018; Real et al., 2019a), object detection (Zoph, Vasudevan et al., 2018) or semantic segmentation (L.-C. Chen, Collins et al., 2018). This confirm that the idea of automated network architecture design is feasible. NAS can be perceived as a subfield of AutoML (Hutter, Kotthoff

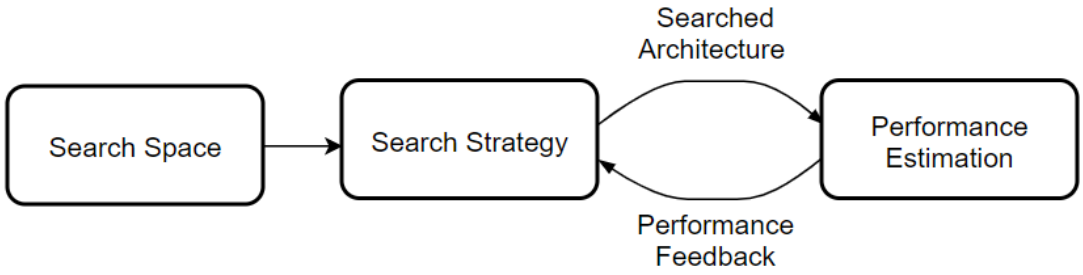


Figure 3.10: Fundamental of Neural Architecture Search procedure.(Elsken, J. H. Metzen, Hutter et al., 2019)

and Vanschoren, 2019) and has significant overlap with hyperparameter optimisation (Feurer and Hutter, 2019) and meta-learning (Vanschoren, 2019).

Methods for NAS will be categorised into three dimensions: search space, search strategy, and performance estimation strategy. As can be seen in Fig 3.10 in NAS technique, a search strategy selects a NAS architecture from a predefined search space. Then, the architecture will be pass to a performance estimation process that returns the performance feedback of the selected architecture to the search strategy. In the following, each section will be explained.

3.6.1 Search Space

The search space determines which neural architectures a NAS approach might discover. A better search space may reduce the complexity of searching for suitable neural architectures. There are different strategies for architecture search spaces. A nearly simple search space includes all sets of layer configurations stacked on each other, as shown in Figure 3.11 (left). This is known as a chain-structured neural network architecture including a sequence of n layers, where the nth layer l_n receives its input from layer n – 1 and its output will be as the input for layer n + 1. Then, the search space is parameterised by the following conditions:

- The number of layers (n)
- The Type of operation for each layer such as convolution, pooling, depth-

wise separable convolutions (Chollet, 2017) or dilated convolutions (F. Yu and Koltun, 2016)

- Hyper-parameters related to the operation such as number of kernels, size of kernel and stride for a convolutional layer (Baker et al., 2017)

Recently, NAS solutions utilise modern element such as skip connections (Zoph, Vasudevan et al., 2018; Real et al., 2019a), which would allow designing more complex and multi-level network as displayed in Figure 3.11 (right). In this case, the input of the layer (i) can be defined as a function that combines all previous layers. These multi-branch architectures have specific cases such as the chain-structured networks, Residual Networks where previous layer outputs are summed (He et al., 2016), or DenseNets, where previous layer outputs are concatenated (G. Huang et al., 2017).

Zoph et al. (2018) propose to search for repeated elements rather than for whole architectures. They optimised two different cells known as a normal cell that keeps the same dimension as the input and reduction cell which decreases the spatial dimension. Then, the final architecture will be created by stacking these cell together (Zoph, Vasudevan et al., 2018), as displayed in Figure 3.12. This search space has some significant advantages in comparison with the search space for the chain structured that will be discussed in the following:

- This search space contains less layer than whole architectures and estimates a seven-times speed-up while attain better performance in comparison with the previous work (Zoph and Le, 2017).
- Neural networks obtained from cells would be adopted to other datasets through changing parameters such as filters or the number of cells.
- In general, designing a neural network by repeating building blocks has established beneficial design principle (e.g. Long short-term memory (LSTM) block in Recurrent Neural Network (RNN)s or stacking a residual block).

The cell-based search space has been successfully applied by some recent research work (Real et al., 2019a; C. Liu, Zoph et al., 2018; H. Liu, Simonyan and Yang, 2019; Elsken, J. H. Metzen and Hutter, 2019). However, the question is how many

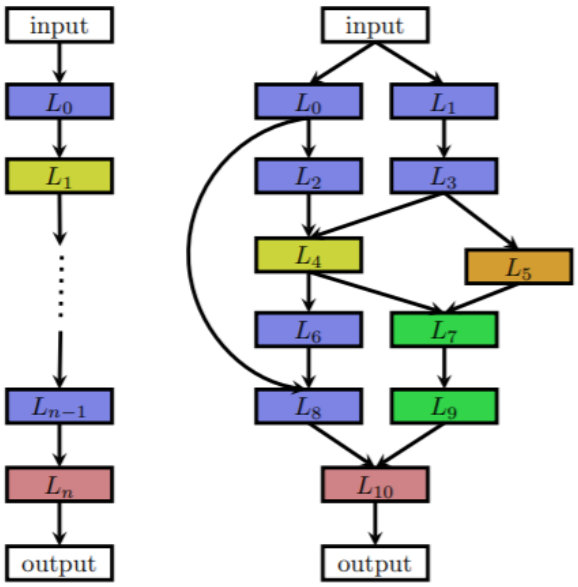


Figure 3.11: An illustration of different architecture spaces. (Elsken, J. H. Metzen, Hutter et al., 2019)

cells should be used or how the cells should be connected to create the actual model. For instance, Zoph et al. (2018) created a sequential design where each cell receives the outputs of the two preceding cells as input.

Liu et al. (2018) introduced the hierarchical search space toward the optimization of architecture that includes three levels of operations. The first level is composed of a set of fundamental operations, the second-level connects the fundamental operations through a directed acyclic graph, and the third level encodes how to connect the second-levels and so on. This thesis adopted the hierarchical search space approach for the view classification and segmentation of LV. In the next section, search strategies that are well-suited for these kinds of search spaces will be reviewed.

3.6.2 Search Strategy

The space of the neural architecture can be investigated with different search strategies such as random search, Bayesian optimization, RL, gradient-based methods, and evolutionary methods.

Bayesian optimization and RL achieved competitive performance on different public datasets such as CIFAR-10 and Penn Treebank benchmarks. However, these search

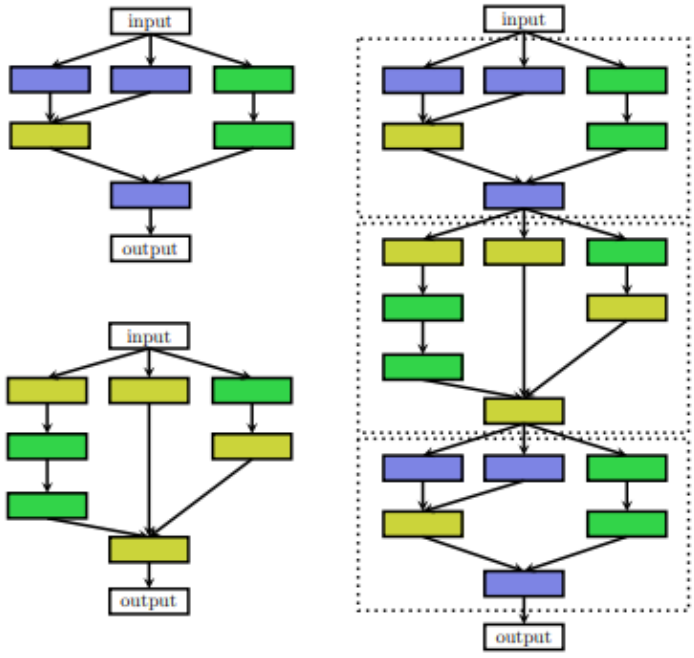


Figure 3.12: An illustration of the cell search space. (Elsken, J. H. Metzen, Hutter et al., 2019)

strategies used immense computational resources to achieve this result (800 GPUs for three to four weeks). After this work, several types of search strategies have been published to decrease the computational costs and obtain further performance improvements (Bergstra, Yamins and Cox, 2013; Zoph and Le, 2017).

To construct NAS as an RL problem (Zoph and Le, 2017; Zoph, Vasudevan et al., 2018), the generation of a neural architecture can be granted to be the agent's action, with the action space identical to the search space. The agent's reward is based on an estimate of the performance of the trained architecture on unseen data. Zoph and Le (2017) employed an RNN policy to sequentially sample a string that in turn encodes the neural architecture. They initially trained this network with the REINFORCE policy gradient algorithm (Williams, 1992).

The first neuro-evolutionary approaches date back to nearly three decades (Miller, Todd and Hegde, 1989) where used genetic algorithms to propose architectures and backpropagation to optimize their weights. Many neuro-evolutionary methods utilize genetic algorithms to optimize the neural architecture and its weights; however, Stochastic Gradient Descent (SGD)-based weight optimization methods currently

outperform evolutionary approaches when comparing to current neural architectures with millions of weights for supervised learning tasks (Elsken, J. H. Metzen, Hutter et al., 2019).

In evolutionary algorithms, a population of the possibly trained network will be developed and in every step, at least one model will be sampled from the population and serves as a parent to generate offspring by applying some local operations such as adding or removing a layer, altering the layer's hyperparameters, adding skip connections, and altering training hyperparameters. After training the offsprings, their performance on a validation set will be evaluated and they will be added to the model(Elsken, J. H. Metzen, Hutter et al., 2019).

A recent case study compared RL, evolution, and the random search and inferring that in terms of final test accuracy, RL and evolution search strategy perform equally well, with the evolution producing better anytime performance and finding smaller models (Real et al., 2019a). This thesis used the evolution search strategy for view classification. In terms of the architecture search method, and evolutionary algorithms would be intensive on the high-resolution images, therefore probably not suitable for semantic image segmentation. Therefore, for the segmentation of LV, a continuous relaxation of the discrete architectures that exactly matches the hierarchical architecture search space has been used.

3.6.3 Performance Estimation Strategy

As discussed in the previous section, the search strategies aim to achieve a neural network that maximizes accuracy or other performance measures on the test dataset. The most manageable way to estimate the performance of the selected network is to train the selected network on training data and assess its performance on validation data. Though, training each network from scratch generates computational demands (thousands of GPU days for NAS). Performance can be estimated based on different strategies such as lower fidelities, learning curve extrapolation, network morphisms (weight inheritance), and one-shot models (weight sharing). In the following, these performance estimation strategies will be discussed:

Lower fidelities performance strategy involves less training times (Zoph, Vasudevan et al., 2018; Zela et al., 2018), training with fewer cells and kernels per layer (Zoph, Vasudevan et al., 2018; Real et al., 2019a), training on lower-resolution images (Chrabaszcz, Loshchilov and Hutter, 2017) or training on a sample of the dataset (Klein et al., 2017). Therefore, in the lower fidelities strategy, computational cost will reduce through the downscale of epochs, data, or models, however, it also includes bias in the evaluation because performance will be underestimated.

Learning curve extrapolation is another performance estimation strategy proposed by (Domhan, Springenberg and Hutter, 2015). Also, Liu et al. (2018a) do not employ the Learning curve extrapolation but proposed predicting performance based on architectural/cell properties and extrapolate to architectures/cells with a larger size than seen during training.

Network morphisms or weight inheritance is another approach to estimating the performance which allows initialising the weights of novel architectures based on weights of other architectures that have been trained before (Wei et al., 2016). This allows increasing the capacity of networks and maintaining high performance without needing training from scratch. Continuing training for a few epochs can also make use of the additional capacity proposed by network morphisms. A benefit of these strategies is that they allow search spaces without an inherent upper bound on the architecture's size (Elsken, J.-H. Metzen and Hutter, 2018). Though, strict network morphisms may lead to complex architectures as it can make architectures larger and this can be attenuated by using approximate network morphisms that support shrinking architectures (Elsken, J. H. Metzen and Hutter, 2019).

One-shot architecture search is another performance estimation strategy that only the weights of a single one-shot model need to be trained, and architectures can then be assessed without any separate training by inheriting trained weights from the one-shot model.

3.7 Neural Architectures Search for Classification

NAS-derived architectures have accomplished highly competitive performance in image classification tasks (Zoph and Le, 2017; Pham et al., 2018; H. Liu, Simonyan and Yang, 2019; Xie et al., 2019). Here, recent popular NAS works on image classification will be described.

Zoph and Le (2016) used a RNN as the controller to compose neural network architectures. Their method is flexible so that it can search variable-length architecture space. They show how the recurrent network can be trained with a policy gradient method to maximize the expected accuracy of the sampled architectures. However, their method needs 800 GPUs for three to four weeks (Zoph and Le, 2017).

Liu et al. (2018) propose a method for learning the structure of CNN based on RL and evolutionary algorithms. Their approach uses a sequential model-based optimisation strategy, in which they search for structures in order of increasing complexity, while simultaneously learning a surrogate model to guide the search through structure space. Direct comparison under the same search space showed that their method is up to 8 times more faster than the RL method of Zoph et al. (2018) in terms of total compute (C. Liu, Zoph et al., 2018).

Real et al. (2019) proposed aging evolution used an evolutionary algorithm known as AmoebaNet-A to discover image classifier architectures. They presented the first controlled comparison of algorithms for image classifier architecture search in a case study of evolution, RL and random search (Real et al., 2019b).

Successful NAS approaches, such as Efficient Neural Architecture Search (ENAS) from Google Brain (Pham et al., 2018) and more recently DARTS (H. Liu, Simonyan and Yang, 2019), have been shown to reduce the search costs by orders of magnitude, requiring ∼100x fewer GPU hours. These methods leverage an important observation that popular CNN architectures often contain repeating blocks or are stacked sequentially. Their effectiveness is thus owing to the key idea of focusing on finding a small optimal computational cell (as the building block of the final architecture), rather than searching for a complete network.

The size of the search space is therefore significantly reduced since the computational cells contain considerably fewer layers than the whole network architecture, which would make such approaches potentially viable for solving real-world challenges.

The DARTS method has been shown to outperform ENAS in terms of the GPU hours required for the search process (H. Liu, Simonyan and Yang, 2019). While most NAS studies report experimental results using standard image datasets such as CIFAR and ImageNet, the effectiveness of DARTS on scientific datasets, including medical images, has also been demonstrated. In this thesis, we have therefore adopted the DARTS method for designing customised architectures to classify echo view images.

3.8 Neural Architectures Search for Segmentation

The majority of NAS-derived networks for image segmentation are designed in encoder-decoder style, exploring the repeatable cells to construct the encoder backbone or the decoder part.

Chen et al. (2018) introduced the first trial on semantic segmentation using NAS. Since DeeLabv3+ has achieved remarkable results, the DeepLab team shifts the emphasis towards the automatic architecture search. Based on the encoder-decoder structure as DeepLab, this work aims to seek a more efficient decoder instead of ASPP following an existing small encoder backbone. A recursive search space is built to encoder multi-scale context information known as Dense Prediction Cell (DPC) (L.-C. Chen, Collins et al., 2018). An efficient random research method is applied as the search strategy. The work achieves 82.7% of the mean IoU (mIoU) on Cityscapes datasets and 87.9% mIoU on PASCAL VOC 12 datasets. Although, the high computational cost (2600 GPU days) limits the application of this approach.

Weng et al. (2019) designed three types of primitive operations set on search space to find two cell architecture automatically for semantic image segmentation especially medical image segmentation. They update the U-Net architectures simultaneously by a differential architecture search strategy. Their proposed model achieved about 0.8 million number of parameters. They demonstrated the good segmentation results on

Promise12, Chaos, and ultrasound nerve datasets, which were collected by magnetic resonance imaging, computed tomography, and ultrasound, respectively (Weng et al., 2019).

Kim et al. (2019) proposed a novel stochastic sampling algorithm based on continuous relaxation for scalable gradient-based optimization on the 3D medical image segmentation tasks. They used four types of cells, encoder-normal, reduction, decoder-normal, and expansion cell to create the encoder and the decoder for the learned 3D U-Net (Çiçek et al., 2016) network (S. Kim et al., 2019).

In another study (Bae et al., 2019) an efficient search space proposed using macro search with parameter sharing for training a controller to apply to 3D medical imaging segmentation tasks. Zhu et al. (2019) examine to search the building blocks to construct U-Net (Ronneberger, Fischer and Brox, 2015) structure for volumetric medical image segmentation problem. They define a cell composed of several convolutional (Conv+BN+ReLU) layers, which are then repeated multiple times to construct the entire neural network. Their segmentation networks follow the encoder-decoder (Milletari, Navab and Ahmadi, 2016; Ronneberger, Fischer and Brox, 2015) structure while the architecture for each cell, (i.e., 2D, 3D, or P3D), is learned in a differentiable way (Z. Zhu et al., 2019).

3.9 Conclusion

This chapter provided some technical background on methods used in this thesis. This included an overview of neural networks, approaches to neural network design, general classification, and segmentation architectures. Moreover, an overview of the NAS including search space, search strategy, and performance estimation strategy has been illustrated. Chapter 4 will apply NAS techniques to design time-efficient and lightweight neural networks for accurate echocardiograhic view classification, while Chapter 5 has focused on designing image segmentation networks.

Chapter 4

Echocardiography View Classification

4.1 Introduction

Echocardiographic (Echo) examinations are typically focused upon protocols containing diverse probe positions and orientations providing several views of the heart anatomy. Standard echo views require imaging the heart from multiple windows. Each window is specified by the transducer position and includes parasternal, apical, subcostal, and suprasternal. The orientation of the echo imaging plane produces views such as long axis, short axis, four-chamber, and five-chamber (Lang, Badano et al., 2015).

Interpretation of echo images begins with view identification. This is a time-consuming and manual process that requires specialised training and is prone to inter-and intra-observer variability. Echo images are very similar and can be particularly challenging for an operator to successfully categorise.

Therefore, accurate automatic classification of heart views has several potential clinical applications such as improving workflow, guiding inexperienced users, reducing inter-user discrepancy, and improving accuracy for high throughput of echo data and subsequent diagnosis.

In most current clinical practice, images from different modalities are managed and stored in PACS. Recently, add-on echo software packages, such as EchoPAC (GE

Healthcare) and QLAB (Philips), attempt to automate the analysis and diagnosis process. However, they still necessitate human involvement in detecting relevant views. Because the automated model could be able to distinguish echo views with distinct characteristics (e.g., PLAX-full and Suprasternal views), but the model may get confused on occasion with no distinct characteristics. For example, the difference of A4CH-LV versus A5CH is whether the scanning plane has been tilted to bring the aortic valve into view, which would make it A5CH. When the valve is only partially in view, or only in view during part of the cardiac cycle. Also, the A3CH view varies from the A2CH view only in a rotation of the probe anticlockwise, again to bring the aortic valve into view. Therefore, integration of information resulting from automated deep learning models and clinician interpretation could be the opportunity to improve the accurac (Narula et al., 2016; Jeganathan et al., 2017; Madani, Arnaout et al., 2018).

As previously stated, echocardiography image frames are not easily discernible by the operator, plus there is often background noise. Therefore, automatic view classification could be widely beneficial for pre-labelling large datasets of unclassified images (Khamis, Zurakhov et al., 2017; J. Zhang et al., 2018).

Application of machine learning algorithms in computer vision has improved the accuracy and time-efficiency of automated image analysis, particularly automated interpretation of medical images (Park et al., 2007; Siegersma et al., 2019; Østvik, Smistad, Aase et al., 2019; S. K. Zhou et al., 2006). However, traditional machine learning methods are constructed using complex processes and tend to have restricted scope and effectiveness (Stoitsis et al., 2006; Doi, 2007). Recent advances in the design and application of deep neural networks have resulted in increased possibilities when automating medical image-based diagnosis (Coates et al., 2013; Tarroni, Bai and Sinclair, 2017).

In this chapter, an overview of the classification methods in the literature will be explained first. This is followed by highlighting the main contribution of this study on the topic of view detection, and a detailed description of the proposed classification models. Finally, the experimental setup, results, and discussion will be presented.

4.2 Previous Work on View Classification

Most previous studies on automatic classification of echocardiographic views have used hand-crafted features and traditional machine learning techniques, achieving varying degrees of success in classifying a limited number of common echocardiographic views (Strubell, Ganesh and McCallum, 2019; Ebadollahi, Chang and Henry Wu, 2004; Agarwal, Shriram and Subramanian, 2013; Hui Wu et al., 2013; Kumar et al., 2010; Otey et al., 2006; Beymer, Syeda-Mahmood and F. Wang, 2008; Kumar et al., 2009; X. Gao et al., 2017).

Following the recent success of deep convolutional neural networks in computer vision, and particularly for image classification tasks, there has been a handful of reports on the application of deep learning for cardiac ultrasound view identification. This section has focused on such studies.

Gao et al. (X. Gao et al., 2017) proposed a fused CNN architecture by integrating a deep learning network along the spatial direction, and a hand-engineered feature network along the temporal dimension. The final classification result for the two-strand-network was obtained through a linear combination of the classification scores obtained from each network. They used a dataset of 432 image sequences acquired from 93 patients. For each strand of the CNN network implemented using Matlab, it took 2 days to process all images. Their model achieved an average accuracy rate of 92.1% when classifying 8 different echocardiographic views.

In another study (Deo et al., 2017), view identification formed part of an automated pipeline designed for the interpretation of echocardiograms. The standard VGG architecture was employed as the CNN model, and 6 different echocardiographic views were included in the study. The class label for each video was assigned by taking the majority decision of predicted view labels on the 10 frames extracted from the video. The overall classification accuracy, calculated from the reported confusion matrix, was 97.7%, and no results for single image classification was reported.

In a follow-up study (J. Zhang et al., 2018), they included 23 views (9 of which were 3 apical planes, each one divided into 'no occlusions', 'occluded LA', and 'occluded LV'

categories) from 277 echocardiograms. The reported overall accuracy of the VGG model dropped to 84% at an individual image level, with the greatest challenge being distinctions among the various apical views. By averaging across multiple images from each video, higher accuracies could be achieved.

Madani et al. (Madani, Arnaout et al., 2018) proposed a CNN model to classify 12 standard B-mode echocardiographic views (15 views, including Doppler modalities) using a dataset of 267 transthoracic studies (90% used for training-validation, and 10% for testing). An inference latency of 21ms per image was achieved for images with a size of 60×80 pixels.

They also reported an average overall accuracy of 91.7% for classifying single frames, compared to an average of 79.4% for expert echocardiographers classifying a subset of the same test images. However, this may not be a fair comparison as the expert humans were given the same downsampled images that were fed into the CNN model, but the human experts are not trained and have no experience of working with such low-resolution images. Later on, they reported an improved classification accuracy of 93.64% by first applying a segmentation stage, where the field of view was extracted from the images using U-net model(Ronneberger, Fischer and Brox, 2015) and the isolated image segment was then fed into the classifier (Madani, Ong et al., 2018).

In a more recent study (Østvik, Smistad, Aase et al., 2019), a CNN model was proposed to balance accuracy and effectiveness. The design was inspired by the Inception (Szegedy, Vanhoucke et al., 2016) and DenseNet (G. Huang et al., 2017) architectures. The performance of the model was examined using a dataset of 2559 image sequences from 265 patients, and overall accuracy of 98.3% was observed for classifying 7 echocardiographic views. The reported inference time was 4.4 ms and 15.9 ms when running the model on the GPU and CPU, respectively, for images with a size of 128×128 pixels.

Vaseli et al. (Vaseli et al., 2019) reported on designing a lightweight model with the knowledge of three state-of-the-art networks (VGG16, DenseNet, and ResNet) for classifying 12 echocardiographic views. Maximum accuracy of 88.1% was observed

using their lightweight models, with a minimum inference time of 52μs for images with a size of 80×80 pixels.

However, the reported accuracies are provided for classifying cine loops and are computed as the average of the predictions for all constituent frames in each cine loop. It is unclear how many frames constituted a cine loop. For a cine loop containing 120 frames (time-window of 2s acquired at 60 frames/s), therefore, an inference time of ≥6.2ms would be required to achieve the reported accuracy. A more rigorous examination of their models also seems necessary and, as apparent from the provided confusion matrices, a great majority of the reported misclassifications, seen as a failure of the models, occurred for parasternal short-axis views.

4.3 Main Contributions

Given our two competing objectives of minimising the neural network size and maximising its prediction accuracy, this study aims to adopt the recent NAS solution of DARTS for designing efficient neural networks. To the best of our knowledge, no other study has applied DARTS to the complex problem of echocardiographic views classification.

This study aimed at including subclasses of a given echocardiographic view and did not focused on subclass-based hierarchical classification (Cerri, Barros and De Carvalho, 2014), and each of the echo views examinations has been considered as a single class. In general, the more numerous the view classes, the more difficult the task of distinguishing the views for the CNN model. This is because if a group of images is considered as a single view in one study and as multiple views in another, those multiple views are likely to be relatively similar in appearance. Perhaps this is one of the primary reasons for the wide range of accuracies (84-97%) reported in the literature. This study will use a private dataset to design customised network architectures for the task of echo view classification.

The input image resolution could potentially impact the classification performance. In case of aggressively downsampled images, the relevant features may in fact be

lost, thus lowering the classification accuracy. On the other hand, unnecessarily large images would result in more computations. Nevertheless, all previous reports considered one particular (but dissimilar in different studies) image resolution, the selection of which was always unexplained. Herein, the impact of different input image resolutions on the performance of the model has been investigated.

The accuracy of deep learning classifiers is largely dependent on the size of high-quality initial training datasets. Collecting an adequate training dataset is often the primary obstacle of many computer vision classification tasks. This could be particularly challenging in medical imaging where the size of training datasets are scarce, e.g. because the images can only be annotated by skilled experts. Hence, it would be advantageous to require less training data. Therefore, the influence of the size of training data on the model's performance for each of the investigated networks examined in this study.

No matter how ingenious the deep learning model, image quality places a ceiling on the reliability of any automated image analysis. Echocardiograms inherently suffer from relatively poor image quality. Therefore, also the impact of image quality on the classification performance has been examined.

In light of the above, the main contributions of this chapter can be summarised as follows:

- Inclusion of 14 different anatomical echocardiographic views (outlined in section 4.4 Figure 4.1); larger than any previous study. Additional cases when only 7 or 5 different views were included to investigate the impact of the number of views on the detection accuracy were also investigated.
- Analysis of three well-known network topologies and of a proposed neural network, obtained from applying NAS techniques to design network topologies with far fewer trainable parameters and comparable/better accuracy for echo view classification.
- Analysis of computational and accuracy performance of the developed models using our large-scale test dataset.

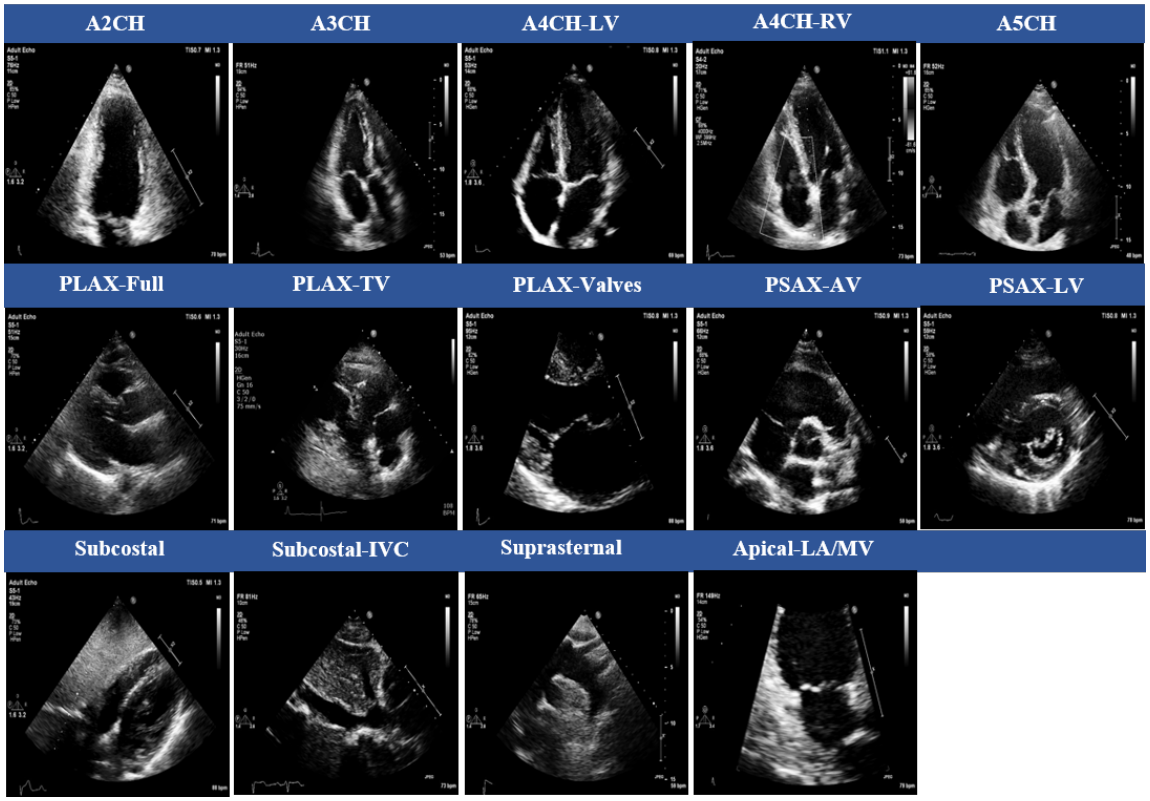


Figure 4.1: The 14 cardiac views in transthoracic echocardiography: apical two-chamber (A2CH), apical three-chamber (A3CH), apical four-chamber left ventricle focused (A4CH-LV), apical four-chamber right ventricle focused (A4CH-RV), apical five-chamber (A5CH), parasternal long-axis (PLAX-Full), parasternal long-axis tricuspid valve focused (PLAX-TV), parasternal long-axis valves focused (PLAX-Valves), parasternal short-axis aortic valve focused (PSAX-AV), parasternal short-axis left ventricle focused (PSAX-LV), subcostal (Subcostal), subcostal view of the inferior vena cava (Subcostal-IVC), suprasternal (Suprasternal), and apical left atrium mitral valve focused (LA/MV).

- Analysis of the impact of the input image resolution; 4 different image sizes were investigated.
- Analysis of the influence of the size of training data on the model's performance for all investigated networks.
- Analysis of the correlation between the image quality and accuracy of the model for view detection.

4.4 PACS-Dataset

This section will introduce the private dataset used for the 2D echocardiographic view classification in this thesis. A random sample of 374 echocardiographic examinations of different patients and performed between 2010 and 2020 was extracted from Imperial College Healthcare NHS Trust's echocardiogram database. The acquisition of the images was performed by experienced echocardiographers and according to standard protocols, using ultrasound equipment from GE and Philips manufacturers.

Ethical approval was obtained from the Health Regulatory Agency (Integrated Research Application System identifier 243023). Only studies with full patient demographic data and without intravenous contrast administration were included. Automated anonymisation was performed to remove all patient-identifiable information.

The videos were annotated manually by an expert cardiologist (JPH), categorising each video into one of 14 classes which are outlined in Figure 4.1. Videos thought to show no identifiable echocardiographic features, or which depicted more than one view, were excluded. Altogether, this resulted in 9,098 echocardiographic videos. Of these, 8,732 (96.0%) videos could be classified as one of the 14 views by the human expert. The remaining 366 videos were not classifiable as a single view, either because the view changed during the video loop, or because the images were completely unrecognisable. The cardiologist's annotations of the videos were used as the GT for all constituent frames of that video.

DICOM-formatted videos were then split into constituent frames, and three frames were randomly selected from each video to represent arbitrary stages of the heart cycle, resulting in 41,321 images. The dataset was then randomly split into training (24791 images), validation (8265 images), and testing (8265 images) sub-datasets in a 60:20:20 ratio. Each sub-datasets contained frames from separate echo studies to maintain sample independence.

The relative distribution of echo view classes labelled by the expert cardiologist is displayed in Figure 4.2 and indicates an imbalanced dataset, with a ratio of 3%

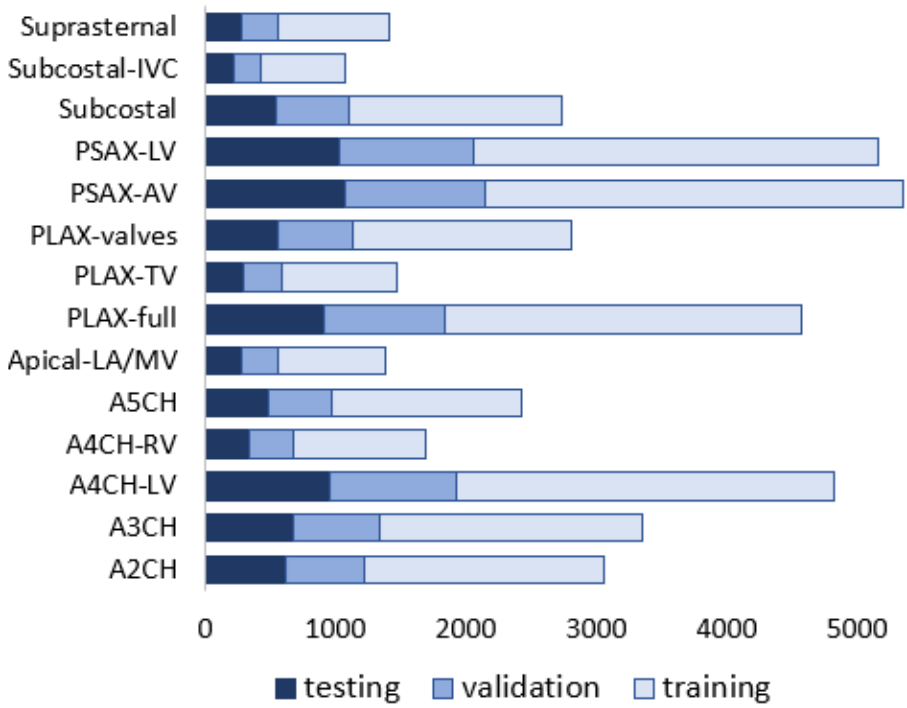


Figure 4.2: Distribution of data in the training, validation and test dataset; values show the number of frames in a given class.

(Subcostal-IVC view as the least represented class) to 13% (PSAX-AV view as the dominant one).

4.5 Method

In this section, a description of a proposed CNN model that has been achieved by the DARTS technique and inspired by the work of Liu et al. (2018) has been provided.

4.5.1 DARTS Method

DARTS method consists of two stages: architecture search and architecture evaluation. Given the input images, it first embarks on an architecture search to explore for a computation cell as the building block of the neural network architecture. Once the best-learnt cell is obtained based on its validation performance, the final architecture could be formed from one cell or a sequential stack of cells. The weights of the cell learned during the search stage are then discarded, and are initialised randomly for the final training stage of the generated neural network model.

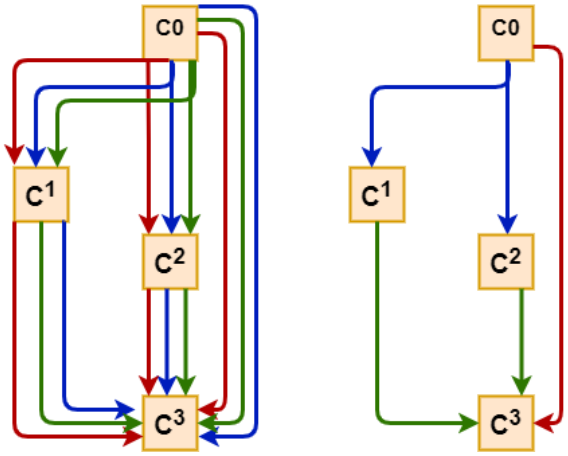


Figure 4.3: Schematic of a DARTS cell. Left: a computational cell with four nodes C^0-C^3. Edges connecting the nodes represent some candidate operations (e.g., 5×5 convolution, 3×3 convolution, and max-pooling represented in Figure 4.3 by red, blue, and green lines, respectively). Right: the best-performing cell learnt from retaining the optimal operations. Figure inspired by (Elsken, J. H. Metzen, Hutter et al., 2019)

Two types of cells are defined in DARTS: (1) a "Normal Cell" which keeps the output spatial dimension the same as input, and (2) a "Reduction Cell" which halves the output spatial dimension while doubling the number of filters/channels.

A cell, depicted in Figure 4.3, is an ordered sequence of N nodes in which one or multiple edges meet. Each node $C^{(i)}$ represents a feature map in convolutional networks. Each edge (i,j) is associated with some operation O(i,j), transforming the node $C^{(i)}$ to $C^{(j)}$. This could be a combination of several operations, such as convolution, max pooling, and ReLU. Each cell is assumed to have 2 inputs which are the outputs from the previous and penultimate cells. The output of the cell is defined as the depth-wise concatenation of all nodes in the cell. The task of learning the optimal cell is effectively finding the optimal placement of operations at the edges.

Each intermediate node $C^{(j)}$ is computed based on all of its predecessors as:

$$C^{(j)} = \sum_{i<j} O^{(i,j)}\left(C^{(i)}\right) \tag{4.1}$$

Refer to equation 4.2, ∂ is a set of candidate operations such as convolution, max pooling where each operation is indicate of some function $O(.)$ to be applied to C(i).

The continuity of the search space is obtained by relaxing the categorical choice of a particular operation to a softmax over all possible operations where for a pair of nodes (i, j), the operation mixing weights are parameterised by a vector $\alpha(i, j)$ of dimension $|\partial|$. Then, the task of architecture search simplify to learning a set of continuous variables $\alpha = \{\ \alpha^(i, j)\}$.

$$\bar{O}^{(i,j)}(C) = \sum_{o \in \partial} \frac{exp({\alpha_o}^{(i,j)})}{\sum_{o' \in \partial} exp(\alpha_{o'}^{(i,j)})} O(C) \tag{4.2}$$

Refer to equation 4.3, at the end of the search, by replacing each mixed operation $\bar{O}^{(\mathrm{i,j})}$ with the most likely operation ($\mathrm{O}(^{(\mathrm{i,j})})$) a discrete architecture can be acquired (edges display in Figure 4.3,right, are the strongest operation). α in 4.3 refers to the architecture. Among all operations, the top 2 strongest operations have been collected, and week operation has been dropped.

$$O^{(i,j)} = argmax_{o \in \partial} \quad \alpha_0^{(i,j)} \tag{4.3}$$

The DARTS method optimises the network weights and associated architecture in conjunction with alternating gradient descent steps on the training data for weights and on validation data for architectural parameters, such as α. The aim for architecture search in DARTS is to identify architecture refers to α^* in (4.4) that minimises the validation loss $\mathcal{L}_{val}(\omega^*, \alpha^*)$, where the weights ω^* associated with the architecture is obtained by minimising the training loss that displays in (4.5). This indicates a bi-level operations problem (Anandalingam and Friesz, 1992; Colson, Marcotte and Savard, 2007) where α is the upper-level variable and ω is the lower-level variable.

$$\min_{\alpha} \quad \mathcal{L}_{val}(\omega^*(\alpha), \alpha) \tag{4.4}$$

$$such.that \quad \omega^*(\alpha) = argmin_{\omega} \quad \mathcal{L}_{train}(\omega, \alpha) \tag{4.5}$$

4.5.2 DARTS Parameters for Architecture Search

For the stage of architecture search, 80% of the dataset was held out for equally-sized training and validation subsets, and 20% for testing. Images were normalised and downsampled to 4 different sizes of 32×32, 64×64, 96×96, and 128×128 pixels, with corresponding batch sizes of 64, 14, 8, and 4, respectively.

The following candidate operations were included in the architecture search stage: 3×3 and 5×5 separable convolutions, 3×3 and 5×5 dilated separable convolutions, 3×3 max-pooling, 3×3 average-pooling, skip-connection, and zero. For the convolutional operations, a ReLU-Conv-BN order was used. If applicable, the operations were of stride one. The convolved feature maps were padded to preserve their spatial size.

A network of 8 cells was then used to conduct the search for a maximum of 30 epochs. The initial number of channels was 16 to make sure the network could fit into a single GPU. Stochastic Gradient Decent (SGD) with a momentum of 0.9, an initial learning rate of 0.1, and weight decay of 3×10^{-4} was used to optimise the weights. To obtain enough learning signal, DARTS utilises zero initialisation for architecture variables indicating the same amount of attention over all possible operations as it is taking the softmax after each operation.

Adam optimiser (Kingma and Ba, 2015) with an initial learning rate of 0.1, momentum of (0.5, 0.999), and weight decay of 10^{-3} were used as the optimiser for α.

4.5.3 Models Training Parameters

Training occurred subsequently, using annotations provided by the expert cardiologist. It was carried out independently for each of the 4 different image sizes of 32×32, 64×64, 96×96, and 128×128 pixels. Identical training, validation, and testing datasets were used in all network models. The validation dataset was used for early stopping to avoid redundant training and overfitting. Each model was trained until the validation loss plateaued. The test dataset was used for the performance

assessment of the final trained models. The DARTS models were kept blind to the test dataset during the stage of architecture search.

Adam optimiser with a learning rate of 10^{-4} and a maximum number of 800 epochs was used for training the models. The cross-entropy loss was used as the network's objective function. For training the DARTS model, a learning rate of 0.1 was deemed to be a better compromise between speed of learning and precision of result and was therefore used. A batch size of 64 or the maximum which could be fitted on the GPU (if <64) was employed.

It is evident from Figure 4.2 that the dataset is fairly imbalanced with unequal distribution of different echo views. To prevent potential biases towards more dominant classes, we used online batch selection where an equal number of samples from each view were randomly drawn (by over-sampling of underrepresented classes). This led to training on a balanced dataset representing all classes in every epoch. An epoch was still defined as the number of iterations required for the network to meet all images in the training dataset.

4.6 Evaluation Metrics

Several metrics were employed to evaluate the performance of the examined and proposed classification models in this study such as classification accuracy, confusion matrix, Macro Average Recall Rate (RECM), and Macro Average Precision (PREM), F measure.

- Classification Accuracy:

  Overall accuracy was calculated as the number of correctly classified images as a fraction of the total number of images which can be computed as follows:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{total\ number\ of\ predictions} \tag{4.6}$$

- Confusion Matrix:

  Confusion matrix is a two-dimensional matrix that presents a brief overview

of the classification performance of a classifier on a test dataset which gives us insight not only into the error being made by a classifier but more importantly the types of errors that were made. In one dimension the true classes of the test dataset, and in the other dimension, the prediction results by the classifier will be assign. In other words, the confusion matrix displays the number of correct and incorrect predictions broken down by each class (Ting, 2010).

- Macro Average Recall (RECM):

  Recall is the fraction of instances of a class that were correctly predicted, and in the binary classification problem recall can be computed by the equation 4.7 where tp (true positive) is where the model correctly predicts the positive class, and fn (false negative) is where the model incorrectly predicts the negative class. For multi-class cases, the macro-averaged recall will be computed by an equation 4.8 which calculates metrics for each class (k) and finds their unweighted mean.

$$REC = \frac{tp}{(tp + fn)} \tag{4.7}$$

$$REC_{macro} = \frac{REC_1 + ... \quad REC_k}{k} \tag{4.8}$$

- Macro Average Precision (PREM):

  Precision is defined as the fraction of correct predictions for a certain class, and in the binary classification case precision can be calculated by the equation 4.9 where tp (true positive) is the correctly predicted samples on the test dataset, and fp (false positive) is the total number of prediction errors. For multi-class cases, macro-averaged precision will be computed by equation 4.10 which gives equal weight to each class and averages the performance of each individual.

$$PRE = \frac{tp}{(tp + fp)} \tag{4.9}$$

$$PRE_{macro} = \frac{PRE_1 + ... \quad PRE_k}{k} \tag{4.10}$$

- F1 Score:

  F1 is an overall measure of a model's accuracy which is the harmonic mean of the precision and recall. A good F1 score means that you have low false positives (fp) and low false negatives (fn), so you're correctly identifying real threats and you are not disturbed by false alarms. Equation 4.11 displays how the F1 score will be calculated. When the F1 score is 1, it is considered perfect, while 0 means the model is a total failure. In the multi-class cases, the F1 score is the average of each class with weighting.

$$F_1 = 2 * \frac{PRE \ * \ REC}{PRE \ + \ REC} \tag{4.11}$$

- Inference Time (latency time):

  Inference time is used to determine the amount of time that a model could return the prediction of view classes for one image from the test dataset on GPU. To this end, a total of 100 images were processed in a loop, and the average time was recorded.

- Number of Parameters and Training Time:

  The number of trainable parameters in each network and training time per epoch was also recorded. All computations were carried using identical hardware resources, allowing for a fair comparison of computational time-efficiency between all investigated network models.

PyTorch (Paszke et al., 2017) was used to implement the models. For the computationally intensive stage of architecture search, a GPU server equipped with 4 NVIDIA Titan V GPUs with 12 GB of memory was rented. For the subsequent training of the searched networks and also the standard models, the utilised GPU was an Nvidia QUADRO M5000 with 8 GB of memory, representing more widely accessible hardware for real-time applications. Inference time (latency time for classifying each

image) was also estimated with the trained models running on the GPU. To this end, a total of 100 images were processed in a loop, and the average time was recorded. All training/prediction computations were carried using identical hardware and software resources, allowing for a fair comparison of computational time-efficiency between all network models investigated in this study.

The number of trainable parameters in the model, as well as the training time per epoch, was also recorded for all CNN networks.

4.7 Experimental Results and Discussion

4.7.1 Architecture Search

The search took ∼6, 23, 42, and 92 hours for image sizes of 32×32, 64×64, 96×96, and 128×128 pixels, respectively, on the computing infrastructure described earlier (section 4.6). Figure 4.4 displays the best convolutional normal and reduction cells obtained for the input image size of 128×128 pixels. The retained operations were 3×3 and 5×5 dilated convolutions, 3×3 max-pooling, and skip-connection. Each cell is assumed to have 2 inputs which are the outputs from the previous and penultimate cells. The output of the cell is defined as the depth-wise concatenation of all nodes in the cell.

Two network architectures were assembled from the optimal cell; "1-cell-DARTS" comprised of one reduction and one normal cell, whereas "2-cell-DARTS" formed from a sequential stack of 2 cells which contain 2 normal and 2 reduction cells. The addition of more cells to the network architecture did not significantly improve the prediction accuracy, as reported in the next section, but increased the number of trainable parameters in the model and thus the inference time for view classification. Therefore, the models with more than 2 cells, i.e. architectures with redundancy, were judged as being comparatively inefficient and thus discarded. Figure 4.4 (left side) also displays the full architecture for the "2-cell-DARTS" model for the input image size of 128×128 pixels.

Results for 5 different network topologies and different image sizes are provided in

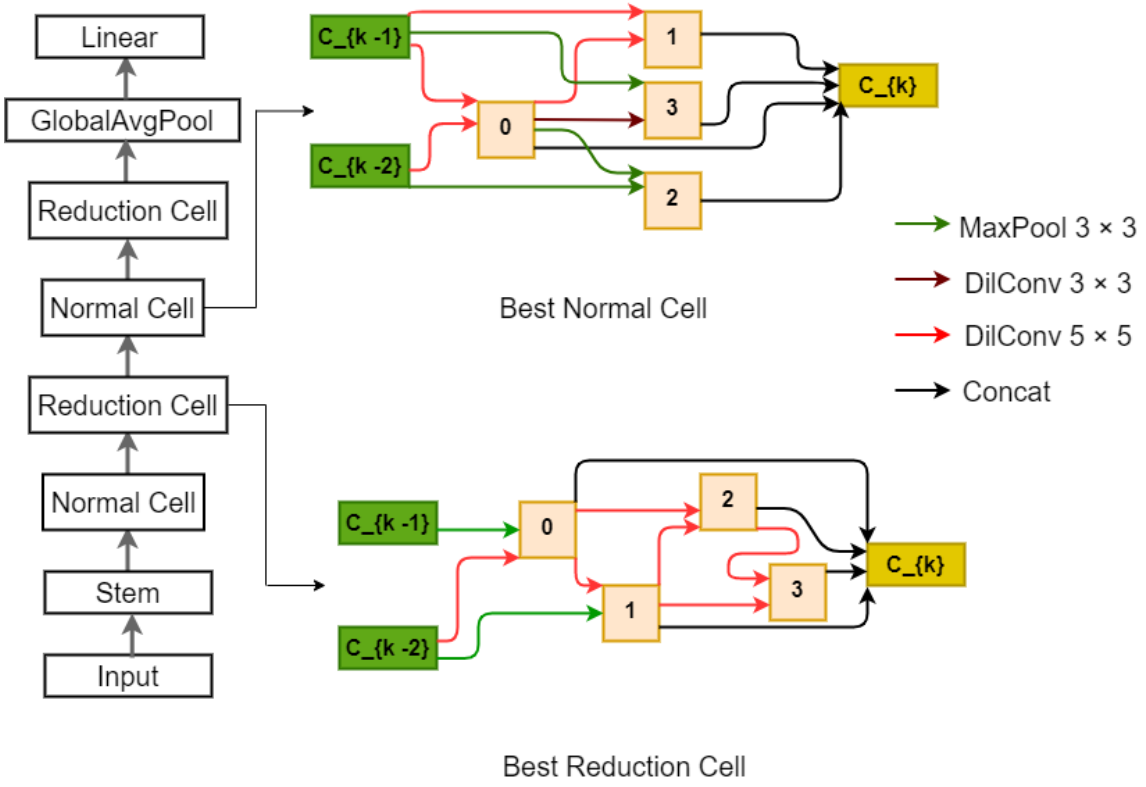


Figure 4.4: Optimal normal and reduction cells for the input image size of 128×128 pixels, as suggested by the DARTS method, where 3×3 and 5×5 dilated separable convolutions, 3×3 max-pooling, and skip-connection operations have been retained from the candidate operations initially included. Each cell has 2 inputs which are the cell outputs in the previous two layers. The output of the cell is defined as the depth-wise concatenation of all nodes in the cell. A schematic view of the "2-cell-DARTS", formed from a sequential stack of 2 cells, is also displayed on the left. Stem layer incorporates a convolution layer and a batch normalisation layer.

Table 4.1. Despite having significantly fewer trainable parameters, the two DARTS models showed competitive results when compared with the standard classification architectures (i.e., VGG16, ResNet18, and DenseNet201). The 2-cell-DARTS model, with only ∼0.5m trainable parameters, achieves the best accuracy (93-96%), precision (92.5-95.2%), and recall (92.3-95.1%) among all networks and across all input image resolutions. Deeper standard neural networks, if employed for echo view detection, would therefore be significantly redundant, with up to 99% redundancy in trainable parameters.

On the other hand, while maintaining a comparable accuracy to standard network topologies, the 1-cell-DARTS model has ≤0.09m trainable parameters and the lowest inference time amongst all models and across different image resolutions (range 3.6-

Table 4.1: Experimental results on the test dataset for input sizes of (32×32), (64×64), (96×96) and (128×128) and different network topologies. Accuracy is the ratio of correctly classified images to the total number of images; precision and recall are the macro average measures (average overall views of per-view measures); F1 score is the harmonic mean of precision and recall. The values in bold indicate the best performance for each measure.

Network	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	Parameters (thousands)	Inference Time (ms)	Time/epoch (s)
				(32×32)			
1-cell-DARTS	88.4	87.8	87.1	87.4	**58**	**3.6**	**41**
2-cell-DARTS	**93.0**	**92.5**	**92.3**	**92.3**	411	7.0	46
ResNet18	90.6	89.9	89.7	89.8	11,177	11.8	184
Vgg16	90.7	89.9	89.5	89.6	134,316	8.3	210
DenseNet201	88.3	87.9	87.0	87.4	20,013	119	1303
				(64×64)			
1-cell-DARTS	90.0	89.4	88.7	89.0	**92**	**6.5**	**81**
2-cell-DARTS	**95.0**	**94.7**	**94.2**	**94.4**	567	12.6	121
ResNet18	92.1	91.5	91.7	91.5		12.0	185
Vgg16	92.4	91.5	92.2	91.8		8.5	240
DenseNet201	93.1	92.5	92.8	92.6		127.3	1322
				(96×96)			
1-cell-DARTS	93.2	92.8	92.3	92.5	**101**	**7.2**	**141**
2-cell-DARTS	**95.4**	**95.1**	**94.9**	**94.9**	669	14.2	264
ResNet18	93.1	92.4	92.2	92.3		12.1	186
Vgg16	93.6	92.9	93.0	92.9		8.6	276
DenseNet201	93.8	93.0	93.3	93.1		129.0	1336
				(128×128)			
1-cell-DARTS	92.5	92.3	91.4	91.8	**89**	**5.9**	**180**
2-cell-DARTS	**96.0**	**95.2**	**95.1**	**95.1**	545	11.8	380*
ResNet18	92.9	92.6	92.2	92.4		12.2	196
Vgg16	93.2	92.1	92.7	92.3		9.0	429*
DenseNet201	93.8	93.1	93.2	93.1		129.4	1605*

* For these experiments, a maximum batch size of <64 could be fitted on the GPU.

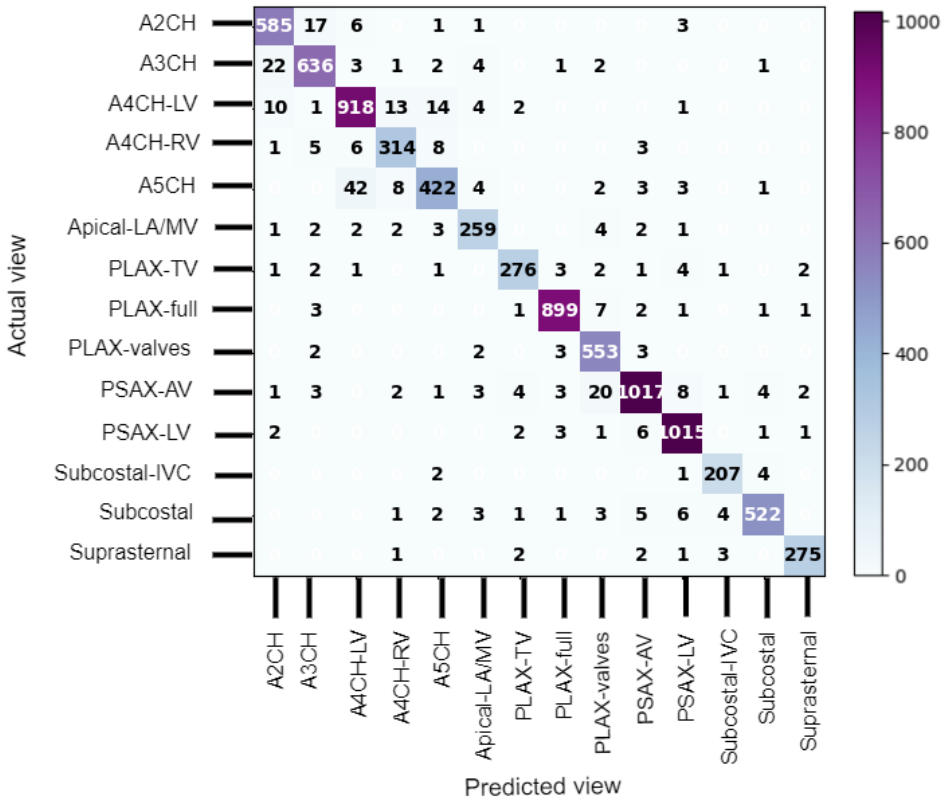


Figure 4.5: Confusion matrix for the 2-cell-DARTS model and input image resolution of 128×128 pixels.

7.2ms). This would allow processing about 140-280 frames per second, thus making real-time echo view classification feasible.

Compared with manual decision making, this is a significant speedup. Although the identification of the echo view by human operators is almost instantaneous (at least for easy cases), the average time for the overall process of displaying/identifying/recording the echo view takes several seconds.

Having fewer trainable parameters, both DARTS models also exhibit faster convergence and shorter training time per epoch than standard deeper network architectures.

4.7.2 View Classification

The confusion matrix for the 2-cell-DARTS model and image resolution of 128×128 pixels is provided in Figure 4.5. The errors appear predominantly clustered between

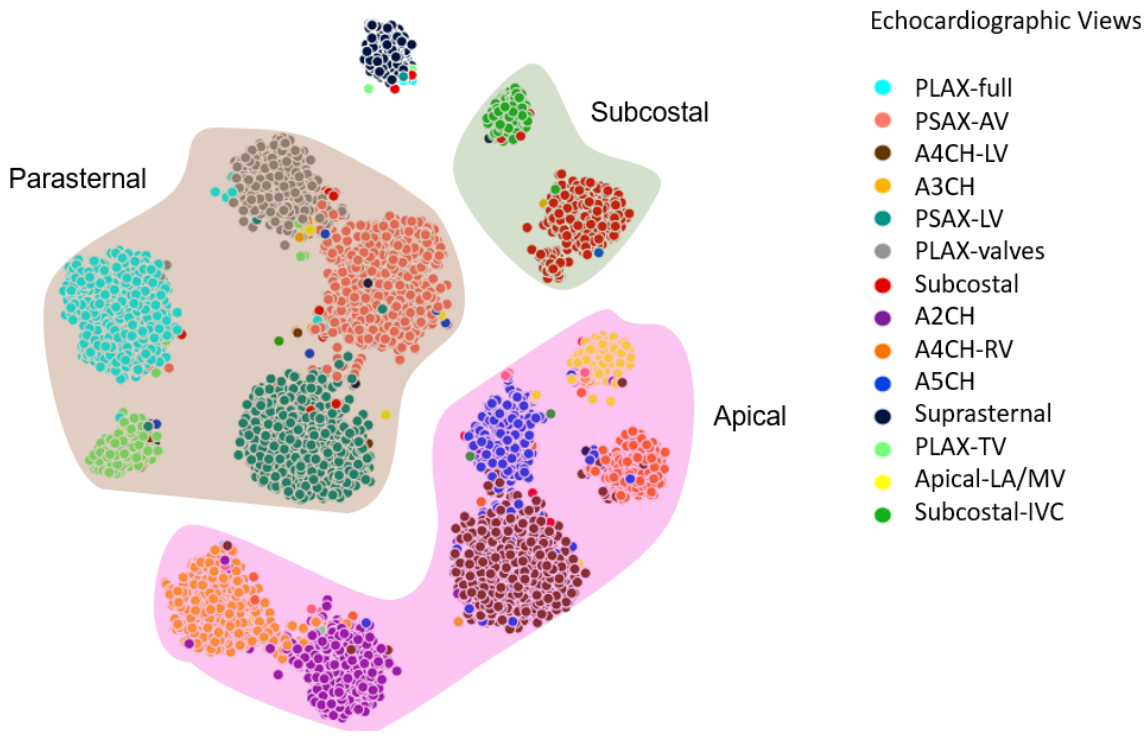


Figure 4.6: t-Distributed Stochastic Neighbor Embedding (t-SNE) visualisation of 14 echo views from the 2-cell-DARTS model (128×128 image size). Each point represents an echo image from the test dataset, and different coloured points represent different echo view classes.

a certain pair of views which represent anatomically adjacent imaging planes. The A5CH view proves to be the hardest one to detect (accuracy of about 80%), as the network is confused between this view and other apical windows. This is in line with previous observations that the greatest challenge lies in distinguishing between the various apical views (Deo et al., 2017).

Interestingly, the two views the model found most difficult to correctly differentiate (A4CH-LV versus A5CH, and A2CH versus A3CH) were also the two views on which the two experts disagreed most often (Howard et al., 2020). The A4CH view is in an anatomical continuity with the A5CH view. The difference is whether the scanning plane has been tilted to bring the aortic valve into view, which would make it A5CH. When the valve is only partially in view, or only in view during part of the cardiac cycle, the decision becomes a judgement call and there is room for disagreement. Similarly, the A3CH view differs from the A2CH view only in a rotation of the probe anticlockwise, again to bring the aortic valve into view.

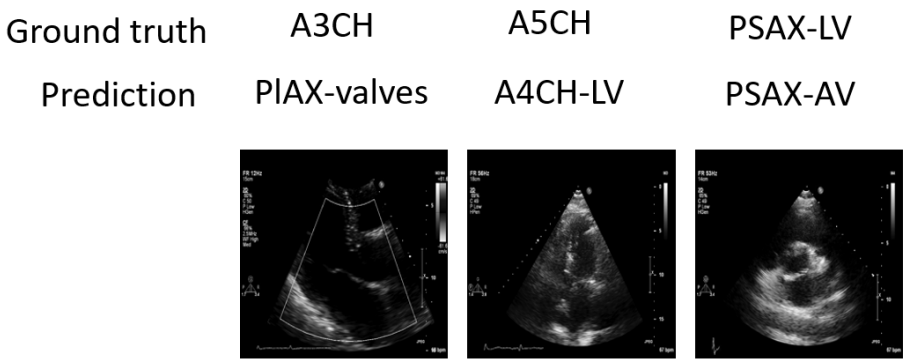


Figure 4.7: Three different misclassified examples predicted by the 2-cell-DARTS model for the image resolution of 128×128 pixels.

It is also interesting to note that the misclassification is not fully asymmetrical. For instance, while 42 cases of A5CH images are confused with A4CH-LV, there are only 14 occasions of A4CH-LV images mistaken for A5CH.

On the other hand, echo views with distinct characteristics are easier for the model to distinguish. For instance, PLAX-full and Suprasternal seem to have higher rates of correct identification, and the network is confused only on one occasion between these two views.

This is also evident on the t-Distributed Stochastic Neighbor Embedding (t-SNE) plot in Figure 4.6, which displays a planar representation of the internal high-dimensional organisation of the 14 trained echo view classes within the network's final hidden layer (i.e. input data of the fully connected layer). Each point in the t-SNE plot represents an echo image from the test dataset.

Noticeably, not only has the network grouped similar images together (a cluster for each view, displayed with different colours), but it has also grouped similar views together (highlighted with a unique background colour). For instance, it has placed A5CH (blue) next to A4CH (dark brown), and indeed there is some "interdigitation" of such cases, e.g. for those whose classification between A4CH and A5CH might be debatable. Similarly, at the top right, the network has discovered that the features of the Subcostal-IVC images (green) are similar to the Subcostal images (red). This shows that the network can point to relationships and organisational patterns efficiently.

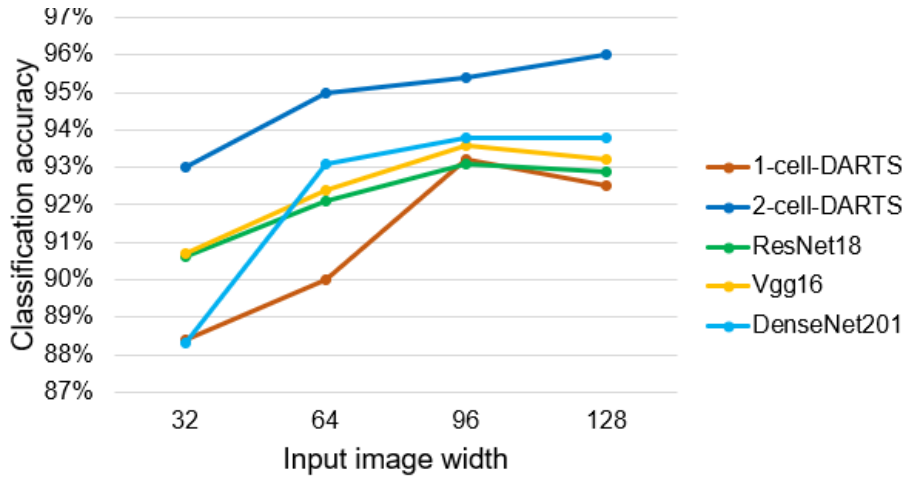


Figure 4.8: Comparison of accuracy for different classification models and different image resolutions; image width of 32 correspond to the image resolution of 32×32 pixels.

Figure 4.7 shows examples of misclassified cases when the prediction of the 2-cell-DARTS model disagreed with the expert annotation. The error can be explained by the inherent difficulty of deciding, even for cardiologist experts, between views that are similar in appearance to human eyes and are in spatial continuity (case of A4CH / A5CH mix-up), images of poor quality (case of A4CH / PSAX mix-up), or views in which a same view-defining structure may be present (case of PSAX-LV / PSAX/AV mix-up).

4.7.3 Impact of Image Resolution, Quality, and Dataset Size

The models seem to exhibit a plateau of accuracy between the two larger image resolutions of 96×96 and 128×128 pixels (Figure 4.8). On the other hand, for the smaller image size of 32×32 pixels, the classification performance seems to suffer across all network models, with a 2.3-5.1% reduction in accuracy relative to the resolution of 96×96 pixels.

Shown in Figure 4.9's upper panel, is the class-wise view detection accuracy for various input image resolutions. Classification accuracy was calculated as the number of correctly classified items as a fraction of the total number of the item. Notably, not all echo views are affected similarly by using lower image resolutions. The drop in overall performance is therefore predominantly caused by a marked decrease in

detection accuracy of only certain views. For instance, A4CH-RV suffers a sharp reduction of ∼25% in prediction accuracy when dealing with images of 32×32 pixels.

Figure 4.9's lower panel shows the relative confusion matrix, illustrating the improvement associated with using image resolution of 96×96 versus 32×32 pixels. Being already a difficult view to detect even in higher resolution images, A5CH will have 47 more cases of misclassified images when using images of 32×32 pixels. Overall, apical views seem to suffer the most from lower resolution images, being mainly misclassified as other apical views. For instance, the two classes associated with the A4CH will primarily be mistaken for one another. This is likely because, with a decreased resolution, the details of the aortic valve would be less discernible by the network.

Conversely, parasternal long-axis views seem to be less affected, and still detectable in downsampled images. For instance, PLAX-full will have only 4 more cases of misclassified images. This might be due to the fact that the relevant features, on which the model relies for identifying this view, are still present and visible to the model.

Overall, and for almost all echo views, the image size of 96×96 pixels appeared to be a good compromise between classification accuracy and computational costs.

To examine the influence of the size of the training dataset on the model's performance, an additional experiment conducted where the training data split into sub-datasets with strict inclusion relationship (i.e., having the current sub-dataset a strict subset of the next sub-dataset), and ensured all the sub-datasets were consistent (i.e., having the same ratio for each echo view as in the original training dataset). Then, all targeted neural networks retrained on these sub-datasets from scratch and investigated how their accuracy varied concerning the size of the dataset used for training the model. The size of the validation and testing datasets, however, remained unchanged.

Figure 4.10 shows a drop in the classification accuracy across all models when smaller sizes of training data are used for training the networks. However, various models are impacted differently. Suffering from redundancy, deeper neural networks require

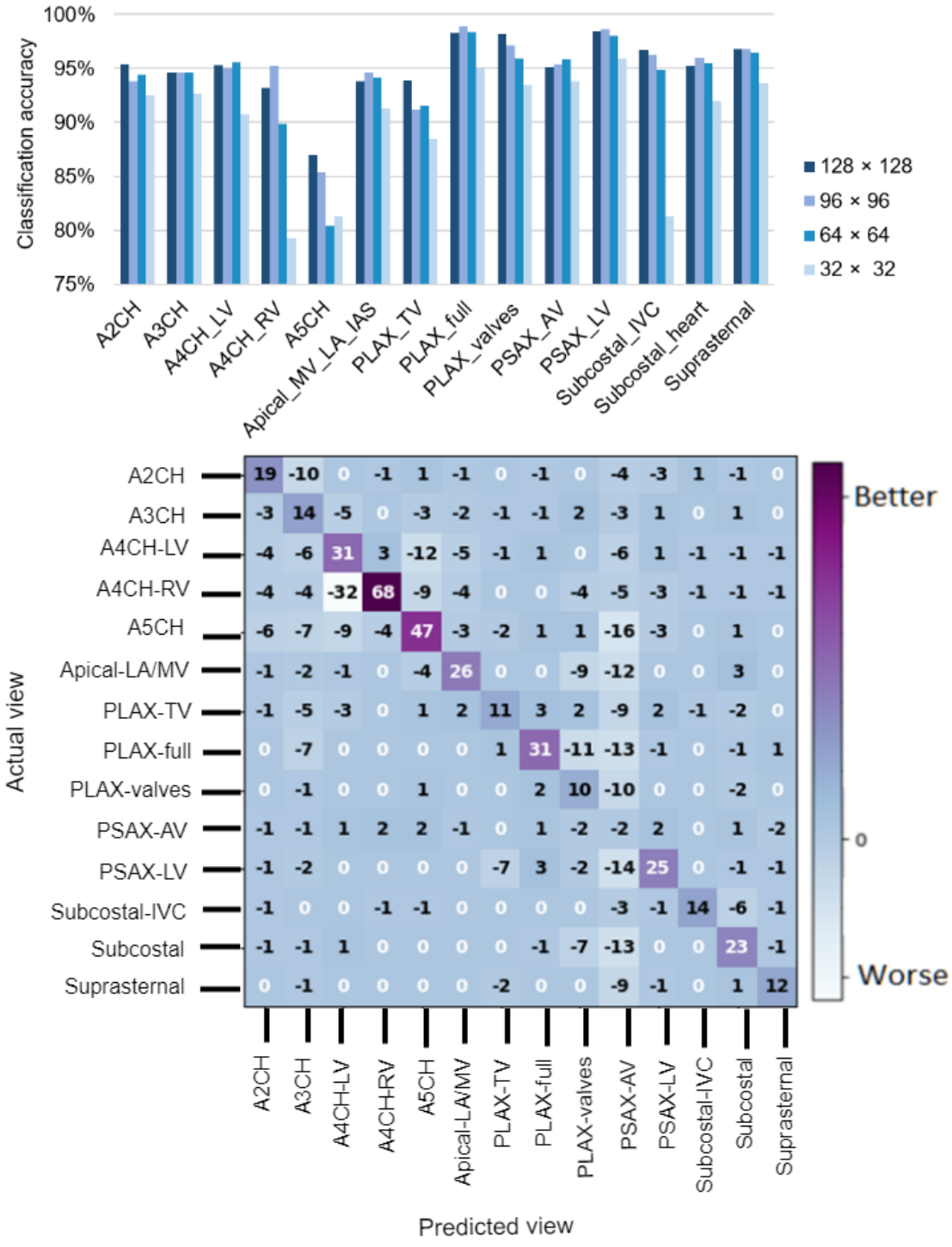


Figure 4.9: Accuracy of the 2-cell-DARTS model for various input image resolutions. Upper: class-wise prediction accuracy. Lower: relative confusion matrix showing improvement associated with using image resolution of 96×96 versus 32×32 pixels.

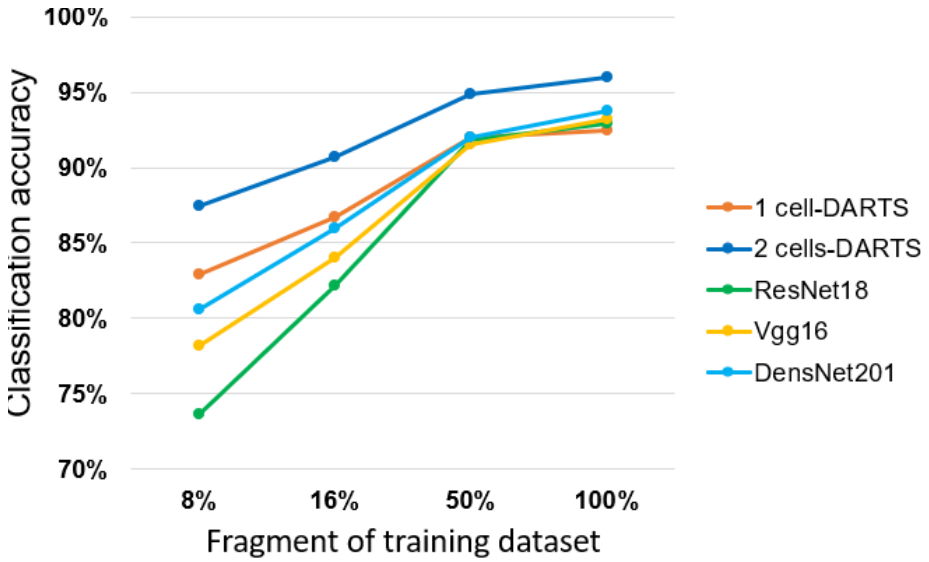


Figure 4.10: Comparison of accuracy of different classification models for an image size of 128×128 versus different fragments of training dataset used when training the models. For each sub-dataset, all models were retrained from scratch.

more training data to achieve similar performances. DenseNet, with the largest number of trainable parameters, appears to be the one which suffers the most, with a 20% reduction in its classification accuracy, when only 8% of the training dataset is used.

However, the DARTS-based models appear to be relatively less profoundly affected by the size of the training dataset, where both models demonstrate no more than an 8% drop in their prediction accuracy when deprived of the full training dataset. When using fewer than 12,400 images (i.e., 50% of the training dataset), both DARTS-based models exhibit superior performance over the deeper networks.

Accordingly, this study hypothesised that the more numerous the echo view classes, the more difficult the task of distinguishing the views for deep learning models, e.g. because of more chances of misclassifications among classes. This is potentially the underlying reason for the inconsistent accuracies (84-97%) reported in the literature when classifying between 6 to 12 different view classes. To investigate this premise, cases when only 5 or 7 different echo views were present in the dataset have been considered. For each study, the aim was including views representing anatomically adjacent or similar imaging planes such as apical windows (thus challenging for the

Table 4.2: The dependence of overall accuracy on the number of echo views; experimental results on the test dataset with 5, 7, and 14 classes for different network topologies, and image resolution of 64×64 pixels. The 7-class study included A2CH, A3CH, A4CH-LV, A5CH, PLAX-full, PSAX-LV, Subcostal-IVC, and a total of 24464 images. The 5-class study included A4CH-LV, PLAX-full, PSAX-AV, Subcostal, Suprasternal, and a total of 18896 images.

Network	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	Parameters (thousands)	Inference Time (ms)	Time/epoch (s)
1-cell-DARTS							
14-classes	90.0	89.4	88.7	89.0	92	**6.5**	81
7-classes	96.4	96.1	96.1	96.1	110	7.8	58
5-classes	**98.1**	**98.3**	**97.9**	**98.1**	**85**	6.6	**38**
2-cell-DARTS							
14-classes	95.0	94.7	94.2	94.4	567	**12.6**	121
7-classes	97.0	96.9	96.7	96.8	709	15.6	85
5-classes	**99.3**	**99.3**	**99.1**	**99.2**	**556**	12.9	**55**

Accuracy is the ratio of correctly classified images to the total number of images; precision and recall are the macro average measures (average overall views of per-view measures); F1 score is the harmonic mean of precision and recall.

models to distinguish), as well as other echo windows. The list of echo views included in each study is provided in Table 4.2.

The results show an increase in the overall prediction accuracy for the two DARTS-based models, when given the task of detecting fewer echo view classes and despite having relatively smaller training datasets to learn from. The 1-cell-DARTS model shows an 8% improvement in its performance when the number of echo views is reduced from 14 to 5. The 2-cell-DARTS model reaches a maximum accuracy of 99.3%, i.e. higher than any previously reported accuracies for echo view classification. This highlights the fact that for a direct comparison of the classification accuracy between the models reported in the literature, the number of different echo windows included in the study must be taken into account.

Finally, to study the impact of image quality on the classification performance, a second expert cardiologist provided an assessment of image quality in the A4CH-LV views and assign a quality label to each image where the quality was classified into 5 grades: very poor, poor, average, good, and excellent. Figure 4.11 displays the

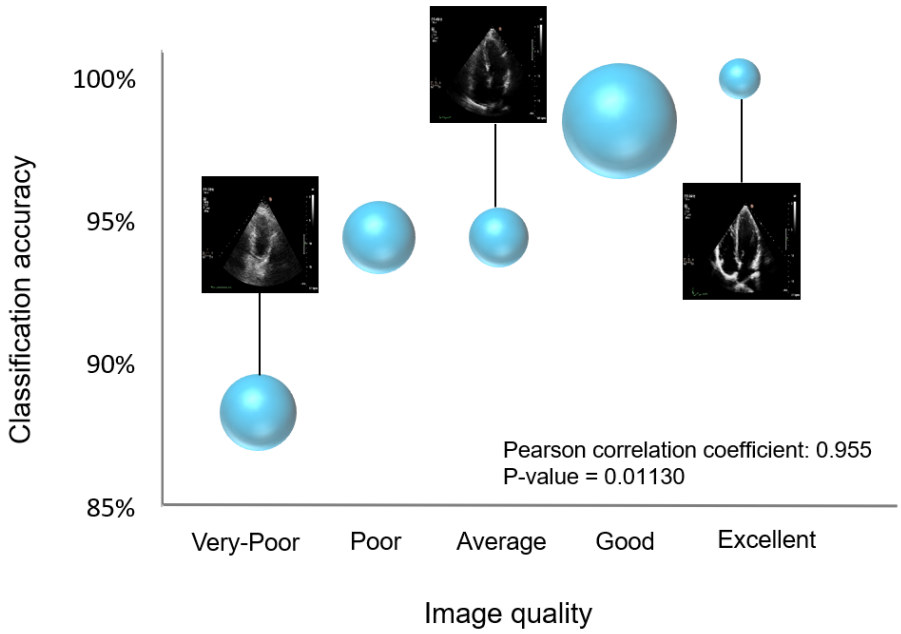


Figure 4.11: Correlation between the classification accuracy and the image quality (judged by the expert cardiologist) of the A4CH-LV view in the test dataset. The area of the bubbles represents the relative frequency of the images in that quality score category. Results correspond to the 2-cell-DARTS model and image resolution of 128×128 pixels.

relationship between the classification accuracy of the 2-cell-DARTS model and the image quality in the test dataset. The area of the bubbles represents the relative frequency of the images in that quality score category, with the "good" category as the dominant grade. This is likely because the image acquisition had been performed mainly by experienced echocardiographers.

The correlation between the classification accuracy and the image quality is evident (p-value of 0.01). Images labelled as having "excellent" quality, indicated the highest classification accuracy of ∼100%. It is apparent that the discrepancy between the model's prediction and the expert annotation is higher in poor quality images. This could potentially be due to the fact that poorly visible chambers with a low degree of endocardial border delineation could result in some views being mistaken for other apical windows.

4.8 Conclusion

In this chapter, efficient CNN architectures are proposed for the automated identification of the 2D echocardiographic views. The DARTS method was used in designing optimised architectures for rapid inference while maintaining high accuracy. A dataset of 14 different echocardiographic views was used for training and testing the proposed models. Compared with the standard classification CNN architectures, the proposed models are faster and achieve comparable classification performance. Such models can thus be used for real-time detection of the standard echo views.

The impact of image quality and size of the training dataset on the efficacy of the models was also investigated. Deeper neural network models, with a large number of redundant trainable parameters, require more training data to achieve similar performances. A direct correlation between the image quality and classification accuracy was observed.

The number of different echo views to be detected has a direct impact on the performance of the deep learning models, and must be taken into account for a fair comparison of classification models. The more numerous the echo view classes, the more difficult the task of distinguishing the views for deep learning models.

Aggressively downsampled images will result in losing relevant features, thus lowering the prediction accuracy. On the other hand, while much larger images may be favoured for some fine-grained applications (e.g., segmentation), their use for echo view classification would offer only slight improvements in performance (if any) at the expense of more processing and memory requirements.

Chapter 5

Left Ventricle Segmentation

5.1 Introduction

Image segmentation is a powerful and challenging technique of image processing to process an image that divides an image into different parts or segments consisting of each section with similar attributes to provides the meaningful objects of the image (Shanazaman, Harapriyasahoo and Rajanjha, 2015; Jaglan, Dass and Duhan, 2019).

Some of the basic applications of image segmentation are object detection, recognition tasks, video surveillance, medical imaging, etc. Two basic types of image segmentation are local segmentation which is concerned with a specific region of an image and global segmentation which is concerned with segmenting the whole image, including a large number of pixels (D. Kaur and Y. Kaur, 2014).

To assess the cardiac function in 2D ultrasound images, quantification of the LV shape and deformation are crucial, and this relies on the accurate segmentation of the LV contour in ED and ES frames (Raynaud et al., 2017). At present, the manual segmentation of the LV suffers from various complications: (i) it needs to be carried out only by an experienced clinician; (ii) inevitable inter-and intra-observer variability in the annotations; (iii) and it is laborious and must be repeated for each patient. Therefore, the automatic segmentation methods may help to resolve these issues and also can lead to increased patient throughput and can reduce the inter-user discrepancy.

There are many suggested methods for 2D LV segmentation. Recently deep convo-

lutional neural networks have been widely applied in medical image segmentation field by extracting features through CNN (Greenspan, Van Ginneken and Summers, 2016; Jafari et al., 2018; Leclerc, Smistad, Pedrosa et al., 2019). In literature, different CNN architectures are constructed to complete the segmentation task such as FC-DenseNet (Jégou et al., 2017), U-Net (Ronneberger, Fischer and Brox, 2015), SegNet (Badrinarayanan, Kendall and Cipolla, 2017).

Same as classification networks, well-designed established segmentation architectures have been developed manually by human experts, which is a time-consuming and error-prone process. Therefore recently, there is significant interest in automated architecture design. In the past years, NAS has successfully recognised neural networks that outperform human-designed architectures on the image classification task (Zoph, Vasudevan et al., 2018; C. Liu, Zoph et al., 2018; Real et al., 2019b). Image classification is a good starting point for NAS; however, image classification should not be the endpoint for NAS, and a recent study shows promise to extend into image segmentation problems (C. Liu, L.-C. Chen et al., 2019). In this study, the Hierarchical Neural Architecture Search method for designing customised segmentation architectures has been adopted.

In this chapter, first, previous work on LV segmentation will be discussed. Followed by the main contribution of this study, dataset, and a detailed description of the proposed segmentation model. Afterward, the experimental setup, results, and discussion will be presented. Finally, this chapter is summarised.

5.2 Previous Work on LV Segmentation

Several studies have been conducted to produce automatic segmentation of LV in echocardiographic images. Traditional methods correspond to methods such as active contour, active shape, and appearance methods, bottom-up approaches, and machine learning-based methods (Noble and Boukerroui, 2006; Carneiro, Nascimento and Freitas, 2011; Oghli, Fallahi et al., 2012; Bosch et al., 2002; S. C. Mitchell et al., 2002; Lin, W. Yu and Duncan, 2003; Wolf et al., 2002; K. E. Leung et al., 2010; Oghli, Mohammadzadeh et al., 2018; Ghelich Oghli et al., 2017). Most of these

methods focus on endocardial border detection in a single frame echocardiography image.

Following the recent success of deep neural networks for medical image analysis (Lu et al., 2017; Anwar et al., 2018; Tajbakhsh et al., 2016), there has been a handful of reports on the application of deep learning for LV segmentation in echo images. This section has focused on such studies.

Carneiro et al. (2011) introduced a new approach based on Deep Belief Network (DBN) that decouples the rigid and nonrigid classifiers to complete the segmentation task of LV in A4C view echocardiographic images. They used a dataset of diseased cases containing 400 annotated images and another dataset of normal cases comprising 80 annotated images, where both sets present long-axis views of the LV. They have compared their segmentation results with two state-of-the-art segmentation models on the dataset of normal cases, and the results show their approach is comparable to the state-of-the-art two approaches (Carneiro, Nascimento and Freitas, 2011).

In another study, a deep learning model proposed to use transfer learning from cross domains to enhance feature representation (H. Chen et al., 2016). Oktay et al. (2017) suggested a strategy that incorporates anatomical prior knowledge and label structure into the U-Net model through a new regularisation model (Oktay et al., 2017).

Smistad et al. (2017) studied if the need for manual annotation can be reduced by using pre-trained U-Net (Ronneberger, Fischer and Brox, 2015) and previously published automatic Kalman Filter (KF) based segmentation model. Their results reveal that CNN can produce similar accuracy to the automatic method, by only training with generated data. The Dice Coefficient (DC) was 0.86 ± 0.06 for the CNN versus 0.87 ± 0.06 (Smistad, Østvik et al., 2017).

Zyuzin et al. (2018) utilised the U-Net network for LV segmentation in echo images in A4C view. They obtained an accuracy of up to 92.3%, which suggests the efficiency of using the U-Net model for automatic identification of the LV endocardial border on echo images (Zyuzin et al., 2018).

Dong et al. (2018) developed a coarse-to-fine framework to complete the segmentation of the LV on 3D echocardiography images. First, they use a deep fusion network and transfer learning, combining the residual modules, to achieve coarse segmentation of LV in 3D. Then, they utilised a geometrical model for a deformable model based on the results of coarse segmentation. Finally, the deformable model was implemented to further optimise the segmentation results (Suyu Dong, G. Luo, K. Wang et al., 2018).

Moradi et al. (2019) proposed a novel architecture that features maps in all levels of the decoder path of U-Net are concatenated, their depths are equalized, and upsampled to a fixed dimension. This stack of feature maps would be the input of the semantic segmentation layer. They evaluated the performance of their proposed model using a private dataset and the public CAMUS dataset introduced in (Leclerc, Smistad, Pedrosa et al., 2019). They achieved an average dice metric of 0.953, Hausdorff Distance (HD) of 3.49 (Moradi et al., 2019) on the CAMUS dataset, and an average dice metric of 0.945 ± 0.12 on the private dataset.

Leclerc et al. (2019) evaluated how far the state-of-the-art encoder-decoder deep CNN methods can go to evaluate 2D echocardiographic images. They introduced two neural networks utilised from the U-Net model called U-Net 1 optimised for speed, and U-Net 2 optimised for accuracy. They used a dataset contains Apical Two-Chamber (A2C) and A4C echo views from 500 patients with annotation from one cardiologist on the full dataset and three cardiologists on a fold of 50 patients. They reveal that encoder-decoder-based architectures outperform state-of-the-art non-deep learning methods (Leclerc, Smistad, Pedrosa et al., 2019). Furthermore, in a more recent study, the same group proposed a multi-stage framework where both the localisation and segmentation steps are optimised simultaneously. Their proposed model is comprised of a combination of the U-Net model followed by a standard regression network. They achieved DC of 0.95 and 0.93 for LV-endo and LV-epi respectively (Leclerc, Smistad, Østvik et al., 2020).

In another study, Azarmehr et al. (2020) examined the performance of U-Net 1 and U-Net 2 introduced by (Leclerc, Smistad, Pedrosa et al., 2019) as well as the original

U-Net model by applying them to an independent dataset of patients to segment the endocardium of the LV in 2D echocardiography images. The prediction outputs of the models are used to evaluate the performance of the models by comparing the automated results against the expert annotations. Their results show that the original U-Net model outperforms other models by achieving an average DC of 0.92 $\pm$ 00.05, and HD of 3.97 $\pm$ 00.82 (Azarmehr et al., 2019).

Li et al. (2020) proposed a deep pyramid and deep supervision network to process each frame of the sequence independently. Their model incorporates a densely connected network, a feature pyramid network, and a deeply supervised network to extract and fuse multilevel and multiscale semantic information. Interestingly, this method outperforms the baseline U-Net model for the segmentation of ED and ES frames (Li, Shizhou Dong et al., 2020). Moreover, later on, part of the same group proposed another method based on a recurrent aggregation network to integrate temporal coherency during the segmentation of one full cardiac cycle. They achieved DC of 0.92 $\pm$0.04 (Li, C. Wang et al., 2020).

5.3 Main Contributions

This chapter aims to adopt NAS algorithm particularly Hierarchical Neural Architecture Search to design a neural network to perform the segmentation of LV with the objective of maximising its prediction accuracy. To the best of our knowledge, no other study has applied NAS technique to the complex problem of echocardiographic LV segmentation.

High-quality and large-scale training datasets are a crucial part of achieving an outstanding deep learning segmentation model. Collecting an adequate training dataset is often the main difficulty of many computer vision segmentation tasks including medical imaging where the size of training datasets is rare, e.g. because the images can only be annotated by skilled experts. Therefore, it would be beneficial to require less training data. Consequently, this study investigates the influence of the size of training data on the model's performance for all developed networks.

Echocardiograms inherently suffer from relatively poor image quality and no matter how ingenious the deep learning model is, the image quality has an impact on the reliability of any automated image analysis. Therefore, the impact of image quality on the performance of the model is investigated. The contributions of this chapter can be listed as follows:

- Adopting the NAS method to design a network for more precise LV segmentation in echo images.
- Inclusion of 2 different private datasets for A4C (outlined in Chapter 2, table 2.1). In addition, including performance reports on a publicly available dataset (CAMUS) which could provide a benchmark for future studies. The public dataset includes A4C and A2C views.
- Comparative performance analysis of four well-known network topologies and the proposed neural network, obtained from applying NAS technique to design network topology with improved accuracy for LV echo segmentation.
- Investigate the influence of the size of the training dataset on the model's performance for the public CAMUS dataset for all investigated networks.
- Investigate the influence of single structure and multi-structure of training data on the model's performance derived from NAS technique.
- Investigate the impact of image quality on the accuracy of LV segmentation model.

5.4 Datasets

This section will present two private and one public dataset known as CAMUS employed for the segmentation in the 2D echocardiographic images.

5.4.1 CAMUS-Dataset

The public CAMUS dataset consists of clinical exams from 450 patients, acquired at the University Hospital of St Etienne (France). The dataset comprises a wide

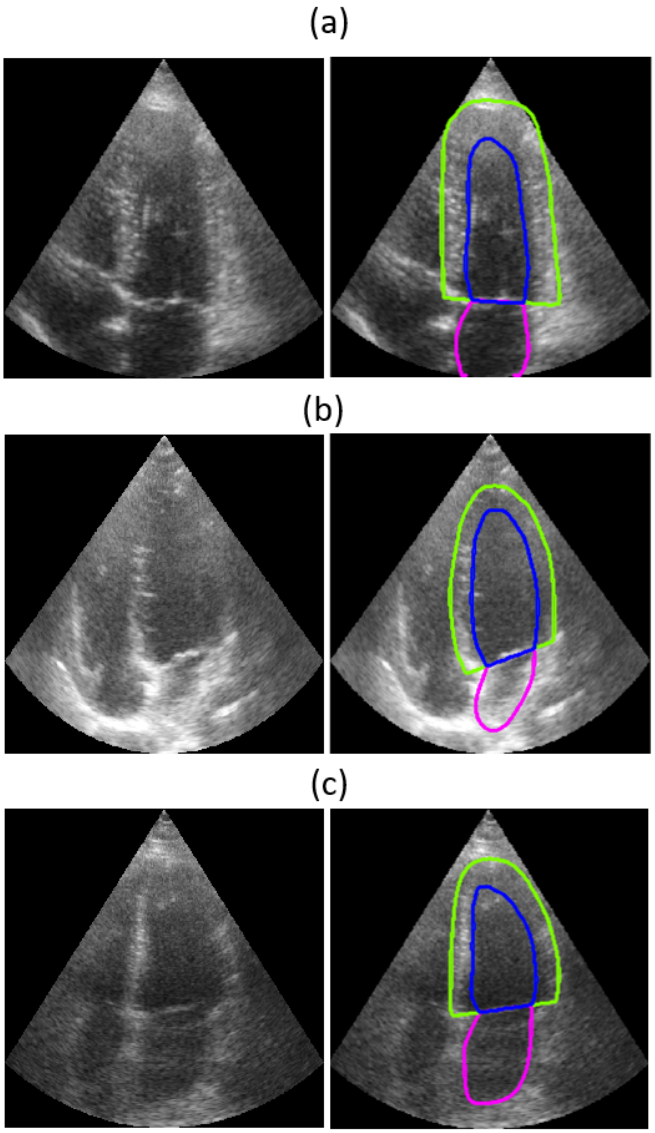


Figure 5.1: Example of images from public CAMUS dataset for (a) Good, (b) Medium and (c) poor image quality. Left: input images; Right: corresponding manual annotations. LV-Endocardium (LV-Endo) and LV-epicardium (LV-Epi) and left atrium (LA) wall are displayed respectively in green, blue and magenta.

variation of acquisition settings. For instance, for some patients, parts of the wall were not visible in the images; this produced a highly heterogeneous dataset, in terms of image quality and pathological cases, which is typical of daily clinical practice data. The full dataset was acquired using GE Vivid E95 ultrasound scanners (GE Vingmed Ultrasound, Horten Norway), with a GE M5S probe (GE Healthcare, US). For each patient, 2D A4C and A2C view sequences were exported from EchoPAC analysis software (GE Vingmed Ultrasound, Horten, Norway). At least one full cardiac cycle

was acquired for each patient in each view, allowing manual annotation of cardiac structures at ED and ES. In total, the dataset includes 1800 2D ultrasound sequences (2 chamber and 4 chamber views of 450 patients) along with the provided multi-structure annotation (i.e. endocardium (LV-Endo), the myocardium (epicardium contour more specifically, named LV-Epi), and the left atrium (LA)) by one expert at the ED and ES instants (Leclerc, Smistad, Pedrosa et al., 2019). An example of images from a public CAMUS dataset with a different range of quality (i.e. good, medium, and poor) has been displayed in Figure 5.1.

5.4.2 PACS-Dataset

PACS dataset is introduced with details in Chapter 4 for echocardiography view classification. From PACS dataset only A4C-LV view has been used for segmentation of LV. From 1606 videos, to obtain the GT (exact solutions) measurements, one experienced cardiologist manually traced the LV borders of 1029 videos. Where the operator judged a video to be of extremely low quality, it was declared invalid and no annotation was made. A custom-made program was developed which closely replicated the interface of echo hardware. The expert visually inspected the cine loops by controlled animation of the loops using arrow keys and manually traced the LV borders using a track-ball for the ED and ES frames. Out of 1029 available videos, a total of 2058 frames were annotated (2 ED/ES frames).

5.4.3 EchoLab-Dataset

The EchoLab private dataset is consisted of 61 patients (30 males), with a mean age of 64±11, who were recruited from patients who had undergone echocardiography with Imperial College Healthcare NHS Trust. Only patients in sinus rhythm were included. No other exclusion criteria were applied. The study was approved by the local ethics committee and written informed consent was obtained. Each patient underwent standard TTE using a commercially available ultrasound machine (Philips iE33, Philips Healthcare, UK), and by experienced echocardiographers. A4C views were obtained in the left lateral decubitus position as per standard clinical guidelines (J. Zhang et al., 2018).

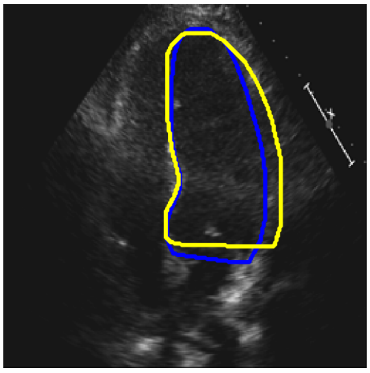


Figure 5.2: An example of A4C view with the LV myocardium segmentation regions overlaid. The blue and yellow curves represent the annotations by Operator-A and Operator-B, respectively.

All recordings were obtained with a constant image resolution of 480×640 pixels. The operators performing the exam were advised to optimise the images as would normally be done in clinical practice. The acquisition period was 10s to make sure at least 3 cardiac cycles were present in all cine loops. In order to take into account, the potential influence of the probe placement (the angle of insonation) on the measurements, the entire process was conducted three times, with the probe removed from the chest and then placed back on the chest optimally between each recording. Therefore, a total of three 10-second 2D cine loops were acquired for each patient. The images were stored digitally for subsequent offline analysis.

To obtain the GT measurements, one accredited and experienced cardiology expert manually traced the LV borders. Where the operator judged a beat to be of extremely low quality, it was declared invalid and no annotation was made. A custom-made program was developed which closely replicated the interface of echo hardware. The expert visually inspected the cine loops by controlled animation of the loops using arrow keys and manually traced the LV borders using a track-ball for the ED and ES frames. Three heartbeats (6 manual traces for ED and ES frames) were measured within each cine loop. Out of 1098 available frames (6 patients × 3 positions × 3 heartbeats × 2 ED/ES frames), a total of 992 frames were annotated. In order to investigate the inter-observer variability, a second operator repeated the LV tracing on 992 frames, blinded to the judgment of the first operator. A typical 2D A4C view

is shown in Figure 5.2, where the locations of manually segmented endocardium by the two operators are highlighted.

5.5 Method

Details of the well-known segmentation network architectures investigated in this study (i.e., U-Net, U-Net ++, SegNet, DeepLabV3ResNet101) can be found in Chapter 3, section 3.5. Here, a detailed description of the designed network will be provided.

Proposed by Liu et al. (2019) Auto-DeepLab use differential NAS to reduce the computational power. Most of the NAS methods usually focus on searching the cell structure and hand-designing an outer network structure. However, Auto-DeepLab will search the network level structure as well as the cell level structure.

5.5.1 Cell Level Search Space

A cell depicted in Figure 5.3,-Bottom is a fully convolutional module consist of B blocks. Each block is a two-branch structure. The set of possible inputs for a block is the output of the previous two cells *(H^{l-2})*, and previous blocks' output *(H^{l-1})* in the current cell (H_1^l, . . . , H_i^l). Therefore, as more blocks are added in the cell, the next block has more choices as a potential source of input. The output of the cell *(H^l)* is the concatenation of output from all of the blocks.

Each block can be defined by a 5-tuple (I_1, I_2, O_1, O_2, C) where I_1, I_2 is the set of all possible selections of input for block i in layer L. O_1, O_2 is a selection of layer types for block C which is the method used to combine the individual outputs of the two branches to get the output of this block (H_I^i).

The types of branches that will be considered in every block are 3×3 depthwise-separable conv, 5×5 depthwise-separable conv, 3×3 atrous conv with rate 2, 5×5 atrous conv with rate 2, 3×3 average pooling, 3×3 max pooling, skip connection and no connection (zero). For the set of possible combination operators *(C)*, element-

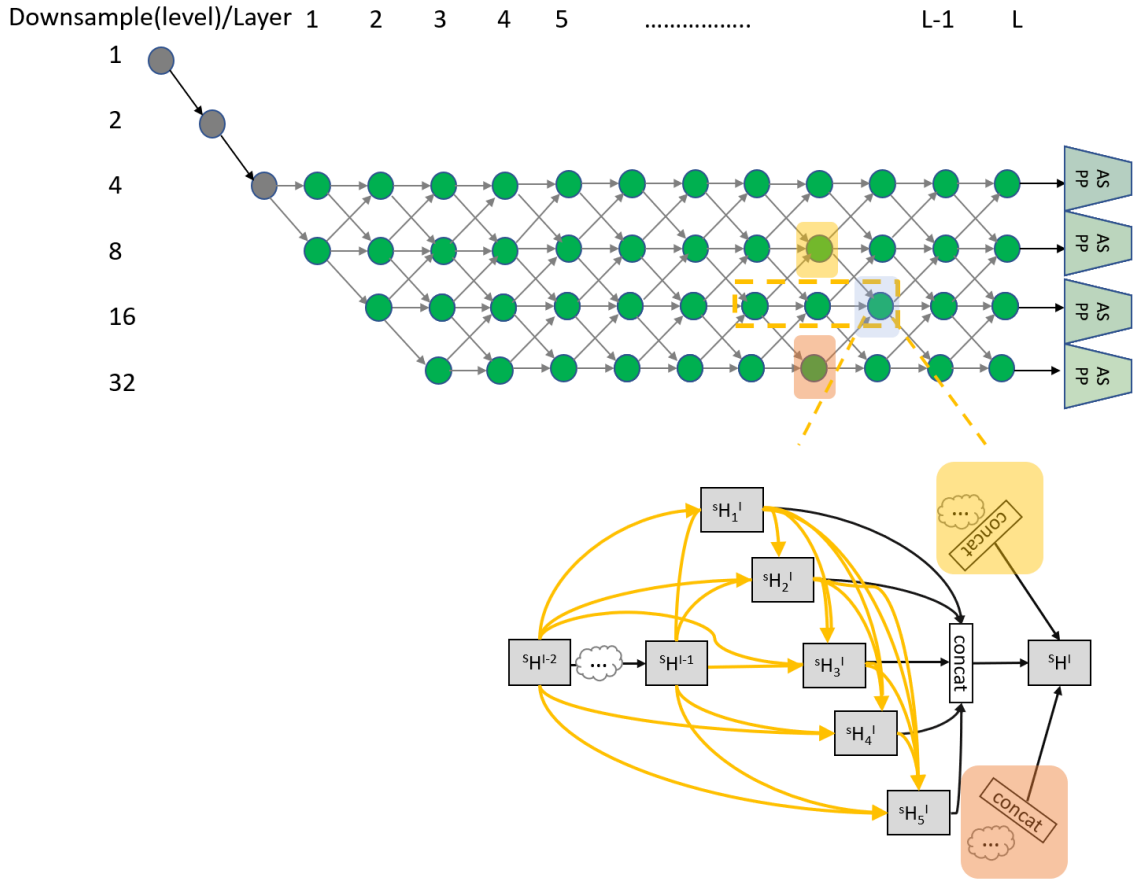


Figure 5.3: Top: network level search space with L = 12. Gray nodes represent the fixed "stem" layers. The path along the green nodes represents a candidate network level architecture. The green dots represent of node (output of each cell) . Bottom: During the search, each cell is a densely connected structure. Every yellow arrow is associated with the set of normalised scalars associated with each operator ($\alpha_{j\to i}$). The three arrows after "concat" are associated with the scalar.

wise addition is the only choice. To form the entire neural network a cell is repeated multiple times.

5.5.2 Network Level Search Space

In the image classification NAS framework, once the best cell structure is found, the entire network is created using a hand design pattern. Therefore, the network level was not part of the architecture search, and the search space for the whole network has never been designed. However, in the dense image prediction problem, the network-level search space is required. The networks for such issues tend to start

with a high-resolution image and get the spatial dimension down somewhere during the network and then back up again to the original dimension.

Among the diverse network architectures for dense image prediction, two principles are consistent: the spatial resolution of the next layer is either twice as large, or twice as small, or remains the same. Also, the smallest spatial resolution is downsampled by 32. Following these common practices, Figure 5.3,-Top displays a mesh of spatial dimension factor versus layer number that is the representation of the network-level search space.

The beginning of the network is a two "stem" structure that each reduces the spatial resolution by a factor of 2. The "stem" layer incorporates a 3×3 convolutions layer and a batch normalisation layer. This “stem” has been shown to be effective for segmentation in (Zhao et al., 2017; P. Wang et al., 2018). Next, there are L layers with unknown spatial resolutions, with the maximum being downsampled by 4 and the minimum being downsampled by 32. Since each layer may differ in spatial resolution by at most 2, the first layer after the "stem" could only be either downsampled by 4 or 8. The goal is to find a good path in this L-layer trellis.

When searching the architecture, the ASPP (explained in Chapter 3-section3.5) module is put at every possible layer according to every possible final spatial dimension factor (i.e. 4,8,16,32). Their outputs are bilinear upsampled to the original resolution before summed to produce the prediction.

5.5.3 Cell and Network Architecture

The continuous relaxation introduced in DARTS technique (H. Liu, Simonyan and Yang, 2019) has been used for the cell architecture. The weights are given to every possible connection that is possible for all the layers (i.e. input, branch and the spatial dimension to be selected) using equations described in the following.

Every block's output tensor (H_i^l) is connected to all hidden states and is computed as equation 5.1 where $(Oj \rightarrow i)$ is possible operator for each branch and (I_i^l) is input tensors.

$$H_i^{\iota} = \sum_{H_j^{\iota} \in I_i^{\iota}} Oj \rightarrow i(H_j^{\iota}) \tag{5.1}$$

Also, each branch $(Oj \rightarrow i)$ is estimated to be used for each connection using differentiable equation 5.2. Along with the normal CNN weights, the meta-weights $(\alpha_{j\rightarrow i}^{k})$ added through the whole latent search space that contains all the connections between possible inputs and the possible branches within a cell. Since equation 5.2 is differentiable, the meta-weights can be trained simultaneously along with the normal weights. After the addition of weights, it will be identified what combination of these meta-weights gives the best structure of the cell.

$$\bar{O}j \rightarrow i(H_j^{\iota}) = \sum_{O^k \in \mathcal{O}} \alpha_j^k \rightarrow iO^k(H_j^{\iota}) \tag{5.2}$$

where

$$\sum_{k=1}^{|\partial|} \alpha_{j\rightarrow i}^{k} = 1 \quad \forall i, j \quad \text{and} \quad \alpha_{j\rightarrow i}^{k} \geq 0 \quad \forall i, j, k \tag{5.3}$$

In other word, $\alpha_{j\rightarrow i}^{k}$ are normalised scalars associated with each operator $O^k \in \mathcal{O}$.

As explained in the cell level search space, each cell has the output from the previous cell *($H^{\iota-1}$)* and the previous two cells *($H^{\iota-2}$)*. Therefore, each output of the cell *(H^{ι})* can be written as equation 5.4 to give output at layer *(L)*:

$$H^{\iota} = Cell(H^{\iota-1}, H^{\iota-2}; \alpha) \tag{5.4}$$

All tensors within a cell with the same spatial size enable the (weighted) sum in Equation 5.1 and Equation 5.2. Though, tensors may take different sizes in the network level as demonstrated in Figure 5.3. Therefore to set up the continuous relaxation, each layer (ι) will have at most 4 hidden states *(${}^{4}H^{\iota}$, ${}^{8}H^{\iota}$, ${}^{16}H^{\iota}$, ${}^{32}H^{\iota}$)*, which the upper left superscript is representative of the spatial resolution.

Also, as explained in the network level search space, the structure of the network needs to be searched. If the spatial dimension factor at the current layer is s, then input to this layer can have three spatial dimensions; twice as small, remain the

same, or twice as large (i.e. $\frac{s}{2}$, s and 2s respectively). A scalar is associated with each solid grey arrows in Figure 5.3, and, the network level update will be defined as:

$$\begin{aligned} {}^{s}H^{\iota} = \beta^{\iota}_{\frac{s}{2}\to s} Cell({}^{\frac{s}{2}}H^{\iota-1}, {}^{s}H^{\iota-2}; \alpha) \\ + \beta^{\iota}_{s\to s} Cell({}^{s}H^{\iota-1}, {}^{s}H^{\iota-2}; \alpha) \\ + \beta^{\iota}_{2s\to s} Cell({}^{2s}H^{\iota-1}, {}^{s}H^{\iota-2}; \alpha) \end{aligned} \tag{5.5}$$

Where s = 4, 8, 16, 32 and ι = 1, 2, . . . , L. Here β is the set of meta-weights which controls the outer network level, therefore depends on the spatial size and layer index. Each scalar in β governs an entire set of α whereas α specifies the same architecture that it is not depends on spatial size or layer index. The relation between β_s is would be given as follows and also implemented as softmax:

$$\beta^{\iota}_{s\to\frac{s}{2}} + \beta^{\iota}_{s\to s} + \beta^{\iota}_{s\to 2s} = 1 \qquad \forall s, \iota \tag{5.6}$$

$$\beta^{\iota}_{s\to\frac{s}{2}} \geq 0 \quad \beta^{\iota}_{s\to s} \geq 0 \quad \beta^{\iota}_{s\to 2s} \geq 0 \qquad \forall s, \iota \tag{5.7}$$

As displayed in Figure 5.3, ASPP modules are connected to each spatial resolution at the L-th layer (atrous rates are adjusted accordingly). Their outputs are bilinear upsampled to the original resolution before summed to produce the prediction.

5.5.4 Optimisation

The scalars control the connection strength between different nodes are part of the differentiable computation graph. Consequently, they can be optimised efficiently using gradient descent. Similar to train the normal weights in CNN, the meta-weights trained using cross-entropy loss to get an optimised condition where various values of α describe the network within a cell and various values of β describe the overall structure. The first-order approximation in DARTS technique (H. Liu, Simonyan

and Yang, 2019) adopted, and the training data partition into two disjoint sets *trainA* and *trainB* to prevent the architecture from overfitting the training data. The optimisation alternates between; update network weights ω by $\nabla\omega\ \mathcal{L}_{trainA}(\omega, \alpha, \beta)$ and update architecture α, β by $\nabla\alpha,\beta\mathcal{L}_{trainB}(\omega, \alpha, \beta)$.

5.5.5 Decoding Architectures

Similar to the DARTS technique, the cell structure was decoded by taking the two strongest connections from all possible inputs using the largest values of α for that branch and repeating this exercise for all the blocks. In addition, "zero" means no connection, and the most likely operator selected by taking the argmax.

For network architecture, the β values that give the maximum value will be selected. Based on equation 5.6, sums of the "outgoing probability" at each of the nodes (green nodes in Figure 5.3) is equal to 1. The β values can be described as the "transition probability" between different spatial resolutions across different layer numbers. The goal is to find the best path with the "maximum probability" from start to end. Then, the path decoded using the classic Viterbi algorithm.

5.5.6 Parameters for Architecture Search

For the stage of architecture search, CAMUS public dataset was used for model development. In this stage, 80% of the public CAMUS dataset was held out for training and validation subsets, and 20% for testing. Half of the train-set images were randomly selected as trainA, and the other half as trainB. Images were normalised and downsampled to 256 $\times$ 256 size with a batch size of 2 due to GPU memory constraint. Stochastic Gradient Decent (SGD) optimiser with a momentum of 0.9, cosine learning rate that decays from 0.1 to 0.000001, and weight decay 0.0003 was used to optimise the weights. The initial values of α, β before softmax are sampled from standard Gaussian times 0.001. They are optimised using Adam optimiser (Kingma and Ba, 2015) with a learning rate of 0.003 and weight decay 0.001.

A total of L = 12 layers in the network and B = 5 blocks in a cell was used to conduct the search. Every green node in Figure 5.3 has B $\times$ F $\times$ $\frac{s}{4}$ output filters,

where B is block, F is the filter multiplier which controls the model capacity and s is downsample rate. During the architecture search, F = 8 is considered. The search was conducted for a maximum of 60 epochs. Also tried searching for longer epochs (100) but did not observe benefit. If α and β were optimised from the beginning when ω are not well trained, the architecture will fall into bad local optima. Therefore, α and β optimised after 30 epochs.

To reduce the spatial size and double the number of filters a stride 2 convolution is used for all $\frac{s}{2} \rightarrow$ s connections. Also, to increase the spatial size and halve the number of filters, bilinear upsampling followed by 1×1 convolution is used for all 2s $\rightarrow$ s connections.

The ASPP module introduced in (L.-C. Chen, Papandreou, Schroff et al., 2017) has 5 branches such as one 1×1 convolution, three 3×3 convolutions with various atrous rates, and pooled image feature. However, in this model ASPP simplified during the search to have 3 branches by only using one 3×3 convolution with atrous rate $\frac{96}{s}$. The number of filters produced by each ASPP branch is still B $\times$ F $\times$ $\frac{s}{4}$.

5.5.7 Models Training Parameters

Training performed on CAMUS, PACS, and EchoLab dataset using annotations provided by the expert cardiologists. The automated model developed through architecture search method using CAMUS dataset have used for training. The training was carried out independently for each of the three different datasets. Identical training, validation, and testing were used in all investigated network models for each dataset. The validation dataset was used for early stopping to avoid redundant training and overfitting. Each model was trained until the validation loss plateaued. The test dataset was used for the performance assessment of the final trained models. The model derived from NAS solution were kept blind to the test dataset during the stage of architecture search.

For training the automated network, a learning rate of 0.1 with a weight decay of 0.0001 and momentum of 0.9, and a maximum number of 6000 epochs was used for training the model. The cross-entropy loss is used for this model. For human-

designed networks, (U-Net, U-Net++, SegNet, DeepLabV3ResNet101) Adam optimiser (Kingma and Ba, 2015) with a learning rate of 0.01 for 6000 epochs deemed to be a better compromise. The Negative log likelihood loss was used as the network's objective function for these networks.

5.6 Evaluation Metrics

Metrics employed to evaluate the performance of the proposed model in segmenting the LV structure are such as the DC, HD, and Intersection-Over-Union (IoU) known as the Jaccard index. Evaluation metrics are defined as follows:

- Dice Coefficient (DC):

  The DC as shown in equation 5.8 was calculated to measure the overlapping regions of the predicted segmentation *(P)* and the GT. The range of DC is a value between 0 and 1, which 0 indicates there is not any overlap between two sets of binary segmentation results while 1, indicates complete overlap.

$$DC = \frac{2|P \bigcap GT|}{|P| + |GT|} \tag{5.8}$$

- Hausdorff Distance (HD):

  The HD was calculated using the equation 5.9 for the contour of segmentation where, *d (j, GT, P)* is the distance from contour point j in GT to the closest contour point in P. The number of pixels on the contour of GT and P is specified with O and M respectively.

$$HD = \max\left(\max_{j\in[0,O-1]} \quad d(j, GT, P),\ \max_{j\in[0,M-1]} \quad d(j, P, GT)\right) \tag{5.9}$$

- Intersection-over-Union (IoU):

  The IoU was calculated image-by-image between the predicted segmentation *(I_p)* and GT. This metric ranges from 0–1 with 0 signifying no overlap and 1

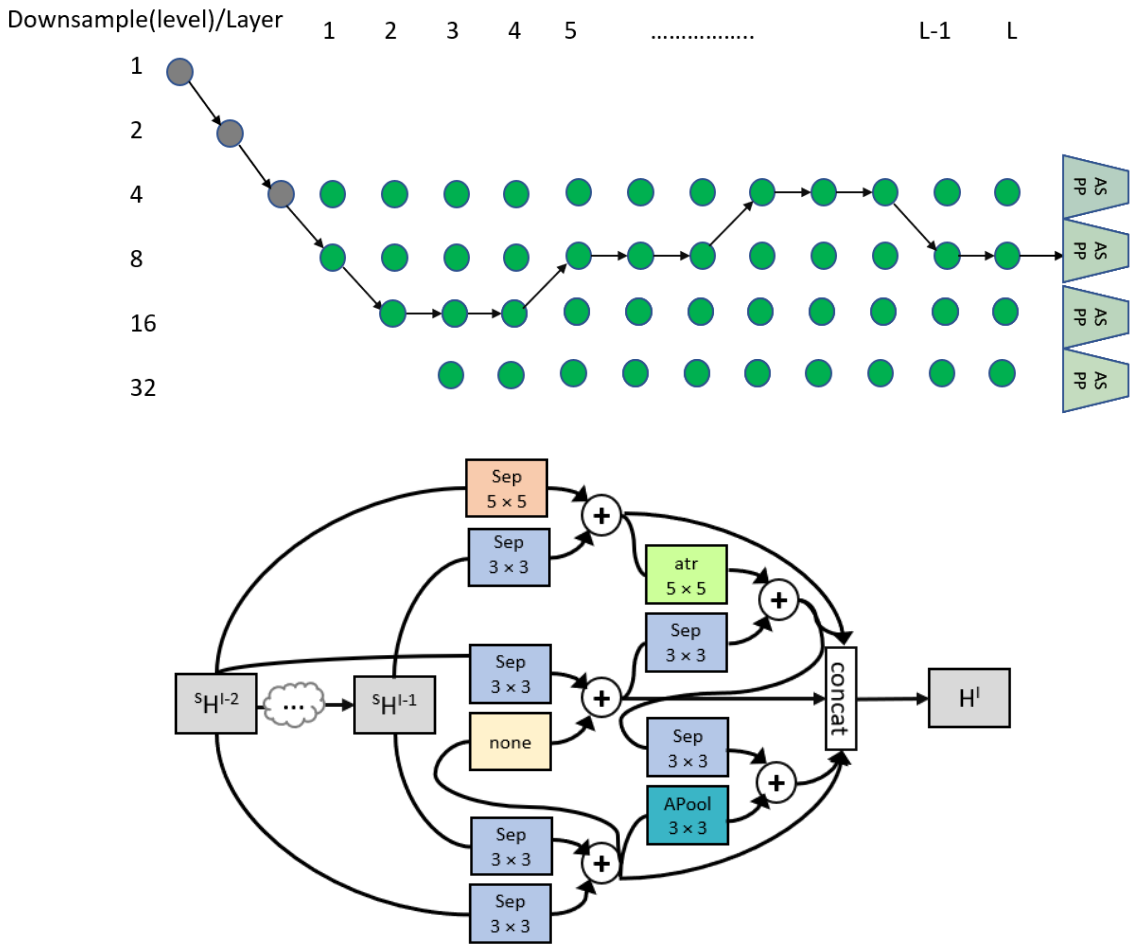


Figure 5.4: Top: The proposed architecture found by the Hierarchical Neural Architecture Search on CAMUS dataset. Bottom: The best cell found for the CAMUS dataset. atr: atrous convolution. Sep: depthwise-separable convolution, none: Zero, and APool: average pooling.

signifying perfectly overlapping segmentation. IoU measures the overlap area between the predicted segmentation and the GT divided by the area of union between the predicted area and GT. IoU is defined as:

$$IoU(GT, I_p) = \frac{|GT \cap I_p|}{|GT \cup I_p|} \tag{5.10}$$

PyTorch (Paszke et al., 2017) was used to implement the models. For the stage of architecture search and also training the models the utilised GPU was an NVIDIA TITAN V GPU with 12 GB of memory. All training/prediction of experiments were carried using identical hardware and software resources.

5.7 Experimental Results and Discussion

All images for all experiments and in all datasets (CAMUS, PACS, and EchoLab) were resized to the dimension of 256×256 pixels, allowing for a fair comparison. All models produce the output with the same spatial size as the input image. All models were trained separately using the annotations provided for each dataset. The DC, HD and IoU were employed to evaluate the performance of the models.

The search took $\sim$ 44 hours for proposed architecture on CAMUS dataset. Figure 5.4 visualises the best cell and network obtained for the proposed model on CAMUS dataset. The retained operations for Auto-Deeplab were 5×5 depthwise-separable conv, 3×3 depthwise-separable conv, no connection (zero), 5×5 atrous conv with rate 2, and 3×3 average pooling. The network is a mesh of spatial dimension factor vs layer number as displayed in figure 5.4. It starts with the high-resolution image and gets the spatial dimension down to a factor of 16 and then back up again to the original dimension by using the ASPP. The total number of layers is 12. In the following, the experimental results for each dataset will be explained.

5.7.1 CAMUS-Dataset

The best cell and network derived from the CAMUS dataset on the search stage have been used to train the CAMUS dataset. The DC, HD and IoU were employed to evaluate the segmentation testing accuracy of the models in segmenting the LV-Endo, LV-Epi, and LA structures on the CAMUS dataset.

Table 5.1 displays the results for five different network topologies on the CAMUS dataset for all three structures. The values in bold correspond to the best values for each metric. From these results, can see that the proposed model achieved from automated neural network design gets the overall best segmentation scores on all metrics for all three structures when compared with the standard dense prediction architectures (i.e. U-Net, U-Net++, SegNet, DeepLabV3ResNet101).

The proposed model achieves the best average DC of (LV-Endo: 0.944, LV-Epi: 0.892, LA: 0.920), HD of (LV-Endo: 2.947, LV-Epi: 2.993, LA: 3.003), and IoU of

Table 5.1: Experimental results on the test public CAMUS dataset and different network topologies. Evaluation measures expressed as mean±SD. The values in bold indicate the best performance for each measure.

Model	DC	HD	iOU
	(LV-Endo)		
U-Net	0.911 ± 0.044	3.388 ± 0.764	0.839 ± 0.069
U-Net ++	0.926 ± 0.041	3.208 ± 0.680	0.864 ± 0.067
SegNet	0.897 ± 0.053	3.558 ± 0.466	0.817 ± 0.082
DeepLabV3ResNet101	0.921 ± 0.048	3.511 ± 0.776	0.856 ± 0.075
Proposed	**0.944 ± 0.034**	**2.947 ± 0.698**	**0.886 ± 0.065**
	(LV-Epi)		
U-Net	0.849 ± 0.053	3.585 ± 0.558	0.741 ± 0.075
U-Net ++	0.859 ± 0.051	3.596 ± 0.501	0.756 ± 0.074
SegNet	0.825± 0.064	3.906 ± 0.524	0.706 ± 0.086
DeepLabV3ResNet101	0.851 ± 0.051	5.128 ± 0.713	0.744 ± 0.074
Proposed	**0.892 ± 0.042**	**2.993 ± 0.604**	**0.784 ± 0.063**
	(LA)		
U-Net	0.887 ± 0.084	4.417 ± 1.709	0.805 ± 0.108
U-Net ++	0.892 ± 0.081	4.166 ± 1.380	0.812 ± 0.103
SegNet	0.849 ± 0.104	5.142 ± 1.322	0.749 ± 0.128
DeepLabV3ResNet101	0.887 ± 0.083	4.693 ± 1.352	0.806 ± 0.111
Proposed	**0.920 ± 0.061**	**3.003 ± 0.670**	**0.833 ± 0.101**

(LV-Endo: 0.886, LV-Epi: 0.784, LA: 0.833) among all networks and for all three structures (i.e. LV-Endo, LV-Epi, LA). As can be seen in Table 5.1, the average of HD has decreased from 3.388 to 2.974, the average of DC has increased from 0.911 to 0.944, and also the average of IoU has increased from 0.839 to 0.886 for the LV-Endo across the U-Net and proposed model. The same condition has been observed for LV-Epi and LA structures. The proposed model outperforms almost other investigated approaches for all three-segmented structures, indicating discrepancies with the GT from which it has learned.

Figure 5.5 displays output examples from the five models on the CAMUS dataset. To specify the borders, the contour of the predicted segmentation was used. The blue is manual annotation (GT) and the red line is prediction. As can be seen, a visual inspection of the automatically detected borders confirms that the proposed model

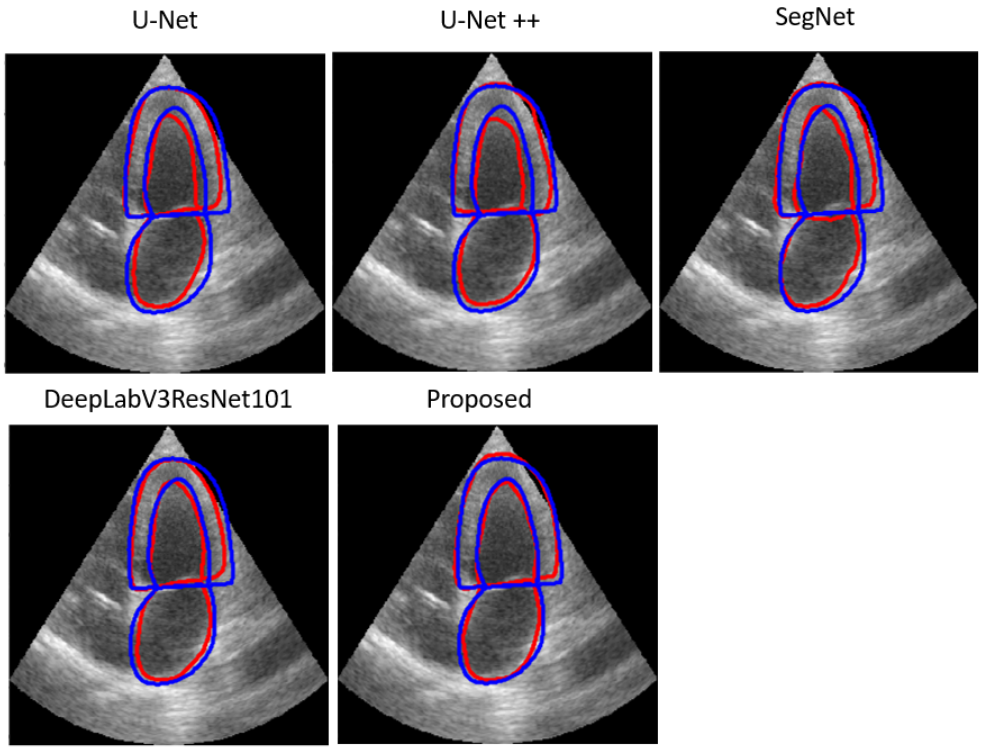


Figure 5.5: Example A4C view outputs from five different models on CAMUS dataset. The prediction is in red while the GT is in blue.

achieved a better result on all the datasets. However, all models look to perform with reasonable accuracy.

Influence of the Size of the Training Dataset

The accuracy of dense prediction networks is dependent on the size of training datasets. Data collection and annotation is a major bottleneck in medical imaging where the size of the training dataset is scarce as the images can only be annotated by experts. Therefore, it would be beneficial to require less training data. This study examined the influence of the size of training data on the model's performance for each of the investigated networks in this chapter.

Figure 5.6 exhibits the influence of the size of the training dataset on the quality of the segmentation of the LV-Endo, LV-Epi, and LA structures. To this aim, 4 different experiments for each model conducted where the same data kept as a test and validation set to allow fair comparison. As for the training set, starting from 8% of the training set, then 16%, and 50% until 100% was reached for the last

examination. As can be seen in Figure 5.6, the overall improvement of DC score can be observed for the three cardiac structures with the increasing training set. The performance of the LA structure seems to suffer most across all network models, except for the proposed model interestingly, the improvement between 8% to 100% training set is quite pronounced (e.g. an increase of average DC for the LA from 0.863 to 0.920).

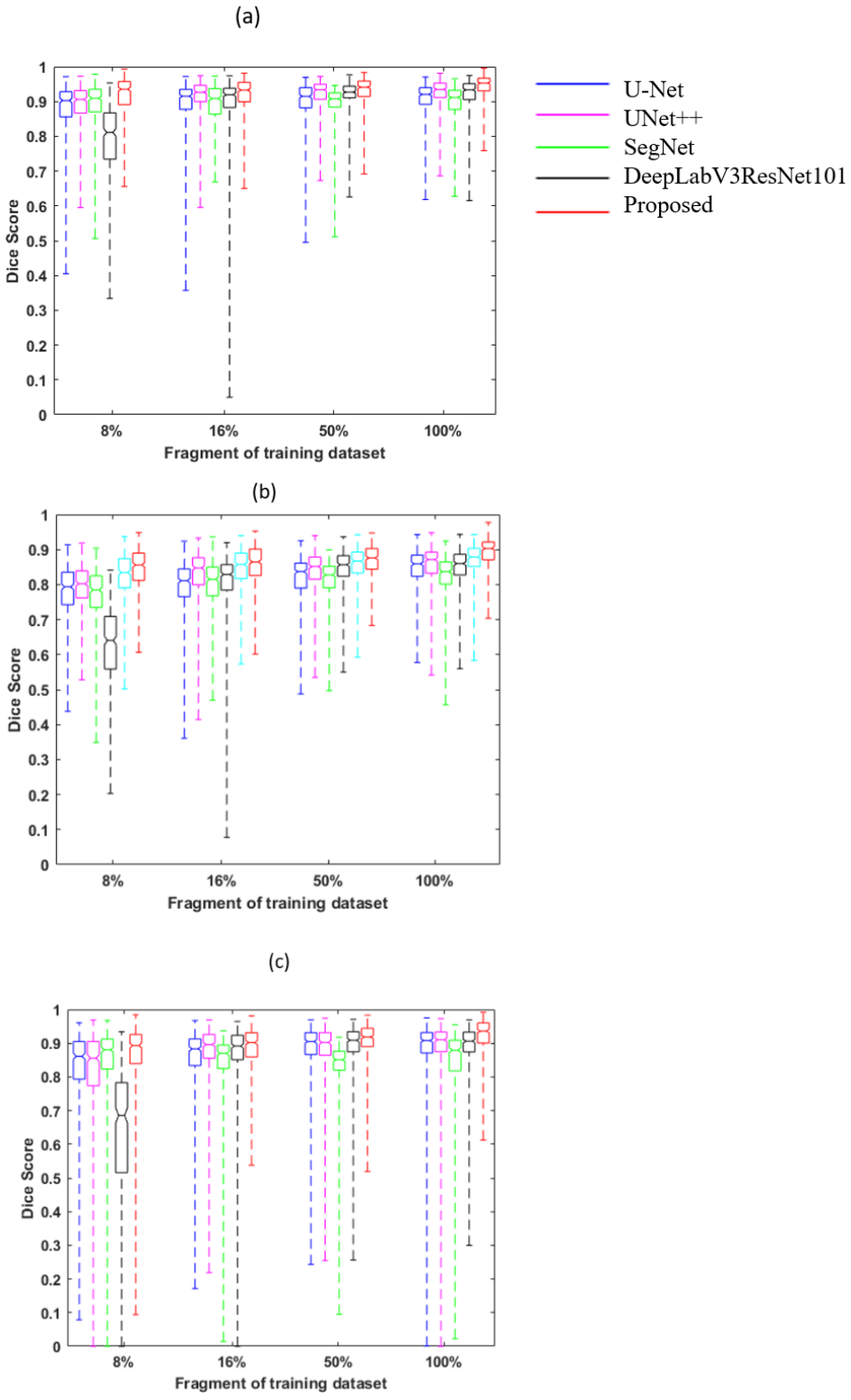


Figure 5.6: Comparison of DC score of different dense prediction models versus different fragments of training dataset used when training the models. (a): LV-Endo, (b): LV-Epi, and (c): LA. For each of the fragments (sub-dataset), all models were retrained from scratch.

The Effect of Poor Quality Images

CAMUS dataset maintains a wide heterogeneity of image quality and pathological cases to preserve the clinical realism. Therefore, to assess the overall robustness of the investigated network including the proposed model, this study has kept the poor image quality in all the experiments as well. Regardless of how intelligent the deep learning model, image quality plays a significant role in the reliability of automated image analysis. Echocardiography images can suffer inherently from poor image quality. Therefore, this study investigated how image quality can affect the performance of the standard dense prediction model in comparison with the proposed model derived from the automated neural network design.

Figure 5.7 in the right panel displays the relationship between the DC score of the U-Net model and the image quality in the test dataset. Also, the left panel demonstrates the relationship between the DC score of the proposed model and the image quality in the test dataset. Section (a), (b), and (c) are relevant to LV-Endo, LV-Epi, and LA structure respectively. The area of the bubbles represents the relative frequency of the images in that quality score category. The "good and medium" categories are the dominant grade.

As can be seen in 5.7-(a) the Pearson correlation coefficient for LV-Endo derived from U-Net and proposed model are 0.851 and 0.990 respectively. This confirms that the U-Net model is more likely to be affected by poor image quality in comparison with the proposed model but the proposed model still shows more robustness on poor image quality cases. However, it is evident that both the U-Net model and the proposed model can cope with the poor-quality images that exist in the CAMUS dataset.

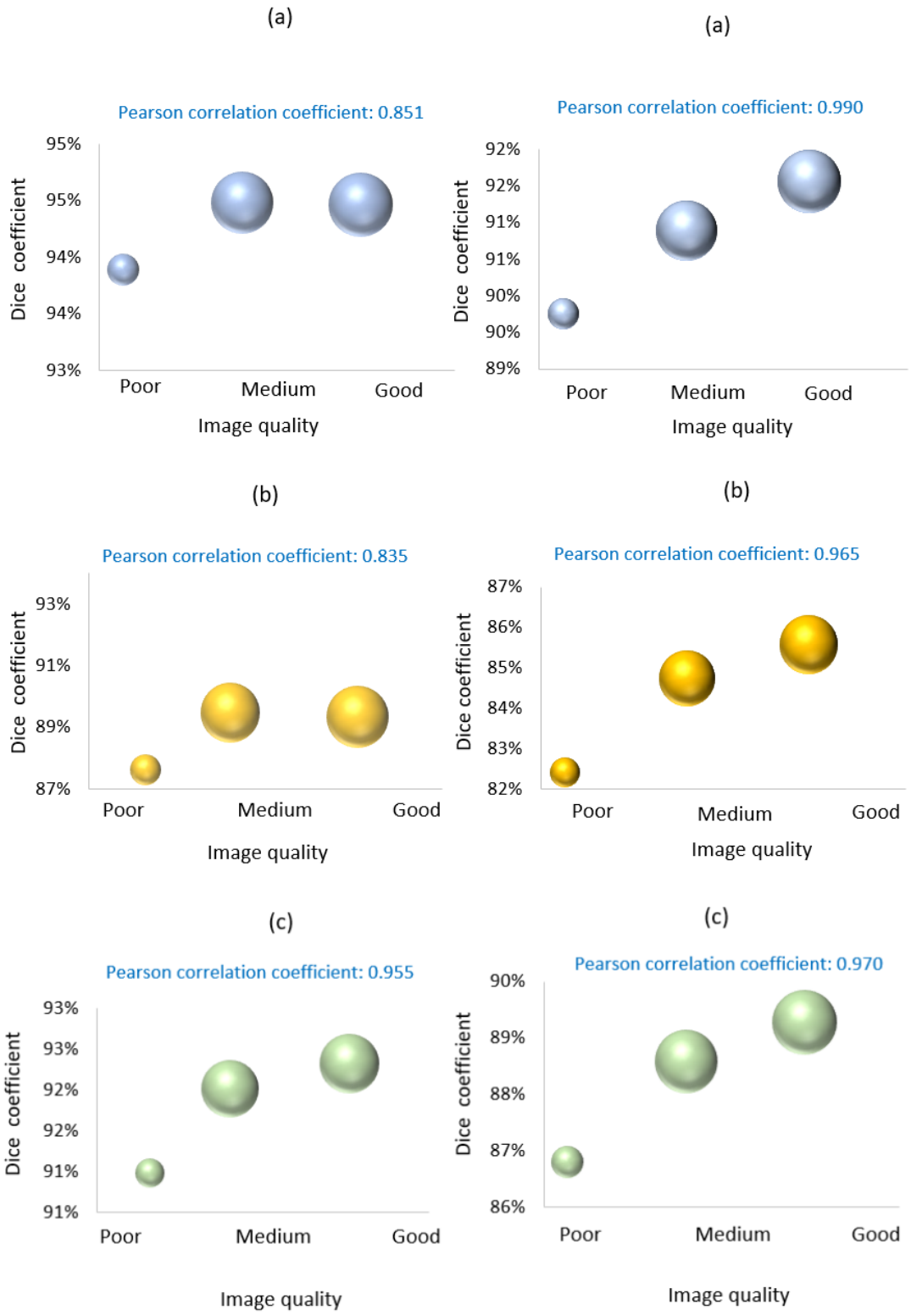


Figure 5.7: Correlation between the dice coefficient score and the image quality (manual quality score provided in CAMUS dataset by the expert) in the test dataset for three structures. (a): LV-Endo, (b): LV-Epi, and (c): LA. Right: Results correspond to the U-Net model, Left: Results correspond to the proposed model. The area of the bubbles represents the relative frequency of the images in that quality score category.

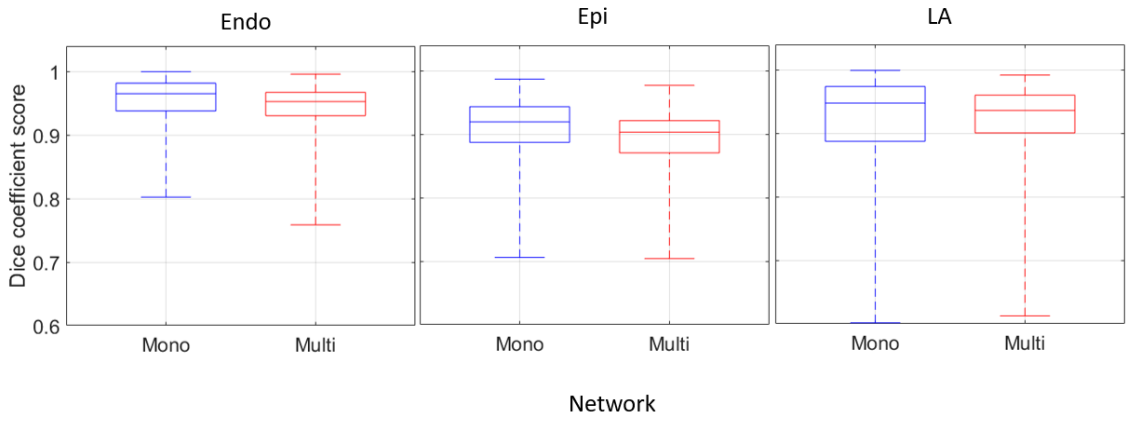


Figure 5.8: Box plots are computed from the results of the proposed architecture for two different approaches. Red boxes for learning to simultaneously segment all three structures (multi). Blue boxes for learning to segment one structure.

Mono Versus Multi-Structures

This study evaluated the impact of learning approaches on the performance of segmentation of the LV-Endo, LV-Epi, and LA. In particular, 4 different experiments have been performed with the same proposed model but with different training sets, i.e. one model trained on only the LV-Endo, one with only LV-Epi, one with only LA, and one on all three structures. Figure 5.8 displays the mono (i.e. trained with only one structure) and multi results of the proposed architecture.

As can be seen in the box plots, the mono and multi-structures approaches produced very close results without considering the structure. This is evident that, with the proposed model, learning the segmentation of only one structure (e.g. LV-Epi) or the combination of structures (e.g. LV-Endo, LV-Epi, and LA) does not improve significantly the results compared to learning the segmentation of the structure alone.

This indicates that the proposed architecture is good enough to exploit the contextual information provided in the segmentation masks. Moreover, even if the segmentation of the LA structure is challenging in comparison to LV-Epi and LV-Endo due to acquisition conditions, the proposed network can get close results both in terms of average DC of 0.954, 0.944, average HD of 2.725, 2.947, and average IoU of 0.894, and 0.886 for mono and multi-structure respectively across the LV-Endo. The same behavior was observed for the LV-Epi and LA structures.

5.7.2 PACS and EchoLab Dataset

The network achieved from CAMUS dataset on search stage has been used to train the PACS and EchoLab datasets separately. The DC, HD and IoU were employed to evaluate the performance of the models in segmenting the LV-Endo region. Table 5.2 provides average DC, HD and IoU for the five different network topologies across PACS and EchoLab datasets for all testing images. The values in bold represent the best scores for the corresponding metric. As for segmentation, the proposed model obtained the best DC of (PACS: 0.943, EchoLab: 0.941), HD of (PACS: 3.694, EchoLab: 4.097), and IoU of (PACS: 0.876, EchoLab: 0.887) on PACS and EchoLab dataset respectively.

Figure 5.9 and 5.10 display output examples from five models for PACS and EchoLab dataset respectively. To specify the LV-Endo border, the contour of the predicted segmentation was used. The solid blue line indicates the manual annotation (GT) while the red line shows the automated results. As can be seen, a visual inspection of the automatically detected borders confirms that the proposed network achieved a better result on all the datasets. However, all models look to perform with reasonable accuracy.

Since for EchoLab dataset, GT was provided with two operators, plausible scenarios for manual or automated (proposed model only) are provided on Table 5.3. For each image, there were 4 assessments of the LV border; 2 human and 2 automated (trained by annotation of either of human operators). The automated model performs similarly to human operators. The model disagrees with Operator-A (OA), but so does Operator-B (OB). Since different experts make different judgments, it is not possible for any automated model to agree with all experts. However, it is desirable for the automated models not to have larger discrepancies when compared with the performance of human judgments; that is, to behave approximately as well as human operators.

Table 5.2: Comparison of segmentation performance between the proposed method and related different network topologies using PACS and EchoLab test dataset. Evaluation measures are expressed as mean±SD. The values in bold indicate the best performance for each measure.

Model	DC	HD	iOU
	(PACS)		
U-Net	0.923 ± 0.048	3.784 ± 1.000	0.861 ± 0.072
U-Net ++	0.927 ± 0.039	3.856 ± 0.992	0.866 ± 0.063
SegNet	0.913 ± 0.046	3.882 ± 0.900	0.843 ± 0.072
DeepLabV3ResNet101	0.911 ± 0.041	3.846 ± 0.999	0.840 ± 0.066
Proposed	**0.943 ± 0.037**	**3.694 ± 0.938**	**0.876 ± 0.064**
	(EchoLab)		
U-Net	0.918 ± 0.047	4.191 ± 0.919	0.852 ± 0.076
U-Net ++	0.916 ± 0.051	4.241 ± 0.981	0.849 ± 0.077
SegNet	0.916 ± 0.093	4.115 ± 0.929	0.854 ± 0.114
DeepLabV3ResNet101	0.931 ± 0.034	4.123 ± 0.895	0.873 ± 0.057
Proposed	**0.941 ± 0.034**	**4.097 ± 0.869**	**0.887 ± 0.060**

Table 5.3: Comparison of evaluation measures expressed as mean±SD for 5 possible scenarios for the proposed model only. OA and OB are Operator-A and Operator-B respectively. POA and POB are the predicted results by the proposed model trained by gold-standard from Operator-A and Operator-B respectively. The values in bold indicate the best performance for each measure.

Model	DC	HD	iOU
	(EchoLab)		
OA vs OB	0.883 ± 0.059	4.496 ± 0.875	0.830 ± 0.030
POA vs OA	**0.941 ± 0.034**	**4.097 ± 0.869**	**0.887 ± 0.060**
POA vs OB	0.926 ± 0.039	4.309 ± 0.964	0.863 ± 0.062
POB vs OB	0.933 ± 0.033	4.551 ± 0.943	0.879 ± 0.055
POB vs OA	0.913 ± 0.060	4.326 ± 0.889	0.862 ± 0.092

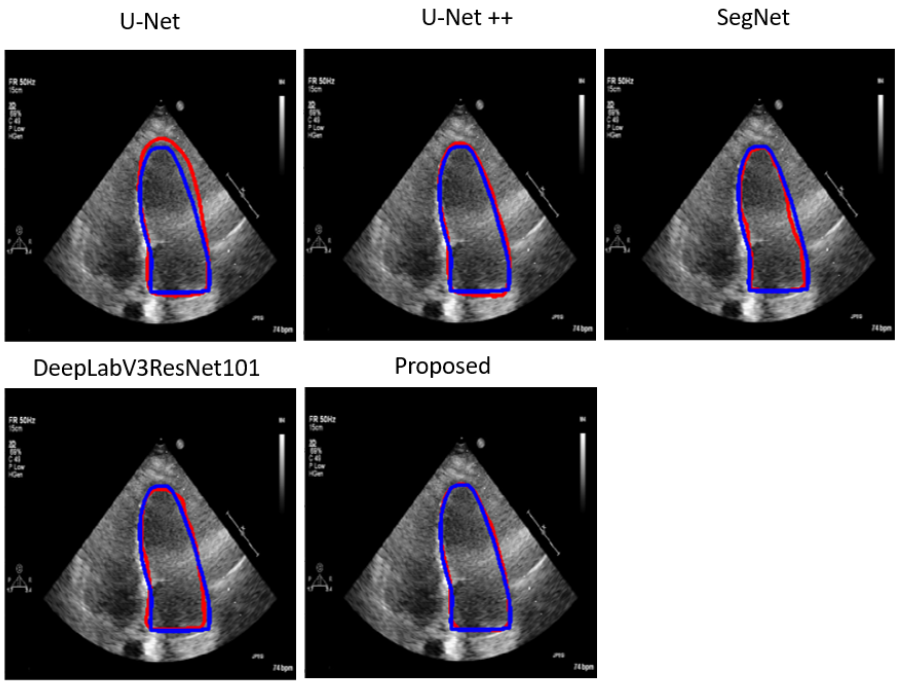


Figure 5.9: Example outputs from five different models on the PACS dataset. In each A4C image, the contours of LV-Endo are displayed. The prediction is in red while the GT is in blue.

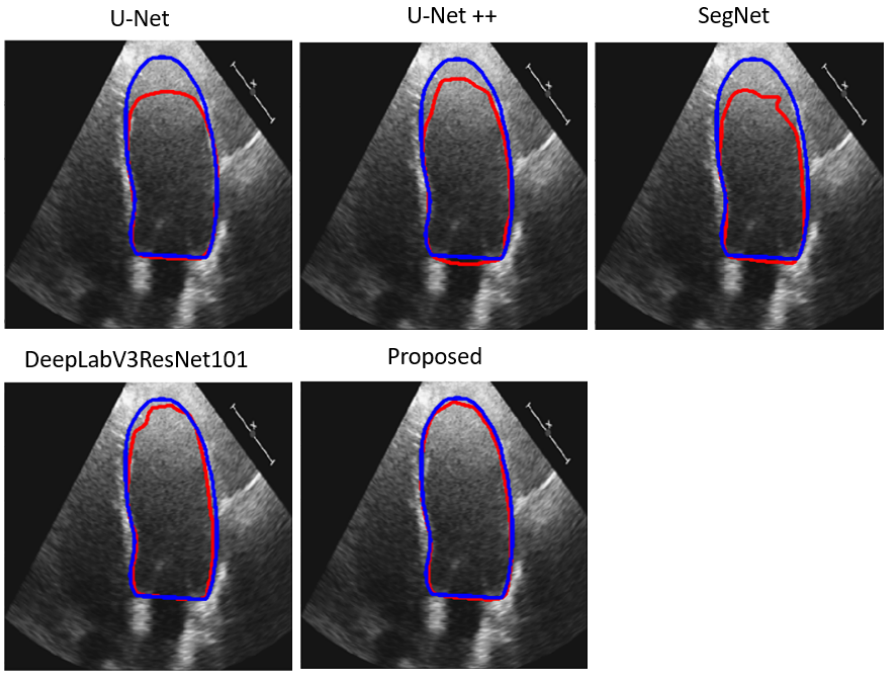


Figure 5.10: Example outputs from five different models on the EchoLab dataset. In each A4C image, the contours of LV-Endo. The prediction is in red while the GT is in blue.

5.8 Summary

In this chapter, the efficient neural network architecture is proposed for the automated segmentation of LV in 2D echocardiographic images. The NAS technique, Hierarchical Neural Architecture Search method, was used to design customised neural networks for segmentation of LV in 2D echocardiographic images. Three different datasets of echocardiographic images including one public and two private datasets were used for training and testing the investigated models. Compared with the established dense prediction architectures, the proposed model achieved comparable performance. The impact of image quality, size of the training dataset, and single or multi-structure on the performance of the networks was also investigated. The proposed model demonstrated robustness on poor image quality. Deeper neural network models require more training data to achieve similar performances. It was also demonestrated that the number of different structures to be segmented has an impact on the performance of the network.

Chapter 6

Speckle Tracking Echocardiography

6.1 Introduction

Two dimensional speckle tracking echocardiography (STE) is a promising relatively new imaging modality. Speckles are created when a random group of scatterers is illuminated by waves bearing a wavelength larger than the size of the individual scatterers. The speckle pattern remains approximately stable from frame to frame. Tracking these speckles frame by frame will allow the extraction of some parameters such as displacement (K. Wu, Shu and Dillenseger, 2014; Jensen, 1996). For example, in a medical application such as evaluation of cardiac function, tracking these speckles and analysing them can help to quantify the myocardial function.

Although there are commercially available STE software packages, the measurements they provide are mutually inconsistent. To address this issue, the EACVI, the ASE along with representatives from all vendors have been endorsing a "task force" aimed to reduce the inter-vendor variability of strain measurement. They propose acceptance in the clinical practice of inter-vendor variability up to 10% (Voigt et al., 2014; James D Thomas and Badano, 2013). However, different commercially available software packages yield unsatisfactorily wide discrepancies between measurements on the same patient images; wider than 10% proposed as acceptable (Voigt et al., 2015).

The processing of ultrasound images is difficult due to typically high levels of noise contained within them. For example, in cardiac ultrasound images tracking walls of the heart is problematic, because of the high level of noise, the lower resolution in

the lateral wall, and the nature of the heart motion. Different approaches for the speckle tracking in ultrasound sequences have been proposed, but it is a complicated task in which there is room for improvement (Tavakoli et al., 2008; Z. Liu and J. Luo, 2017; Albinsson et al., 2018).

Several models have received extensive attention in the ultrasound engineering community, such as BM (Khamis, Shimoni et al., 2016; Zolgharni et al., 2017; De Luca, Székely and Tanner, 2015; Jasaityte, Heyde and D'hooge, 2013), optical flow (Torkashvand, Behnam and Sani, 2012; Porée et al., 2018; Tenbrinck et al., 2013), elastic registration (Chakraborty et al., 2018; Heyde et al., 2012), and machine learning models (Gandhi et al., 2018; Alsharqi et al., 2018). The most computationally efficient method of quantifying tissue motion on ultrasound is BM.

Traditional BM approaches are extremely vulnerable to the presence of image noise, which is always present in everyday clinical cine loops (Voigt et al., 2014). Since BM possesses conceptual simplicity and high computational speed and can provide a robust estimation of the motion by comparing the similarity between blocks of two images or two video frames, it has been commonly used in the ultrasound community (Tavakoli et al., 2008; J.-N. Kim et al., 2002).

This chapter, therefore, investigates methods of improving speckle tracking techniques to enhance its reliability for the calculation of myocardial velocities and deformation parameters such as strain and strain rate. An overview of the speckle tracking studies in the literature will be provided first. This is followed by highlighting the main contributions of the chapter on the topic of speckle tracking, and a detailed description of the proposed BM model. Finally, the experimental setup, results, and discussion will be presented.

6.2 Previous Work on Speckle Tracking

Several studies have attempted to improve the accuracy of the speckle tracking algorithms. Active shape models have been used to extract several physical properties of the myocardium in its different layers by applying some constraints to improve the

accuracy of the motion estimation (D'hooge, Schlegel et al., 2001). The use of the Viterbi algorithm to overcome the effect of peak hopping error has also been reported. They tried to overcome the limitations of speckle decorrelation noise (Petrank, L. Huang and O'Donnell, 2009).

Barbosa et al. (2014) presented a combined method for segmentation and tracking of LV in 4D ultrasound sequences. They used a combination of automatic segmentation at the ED frame and tracking using a global optical flow-based tracker and local BM. The core novelty of the proposed model relies on the recursive formulation of BM. Their proposed model obtained the average segmentation errors of 2.29 and 2.26 mm while the tracking error is not reported (Barbosa et al., 2014).

Khamis et al. (2016) introduced a novel algorithm known as K-SAD that integrates the physiological constraint of smoothness of the displacement field into the tracking algorithm to overcome the limitation of speckle decorrelation noise. They observe improve performance under noisy conditions by comparing a subgroup of 40 subjects with the best image quality. The K-SAD model hardly requires any "post-tracking" techniques that may have positive effect regarding inter-vendor differences (Khamis, Shimoni et al., 2016).

Joos et al. (2018) proposed a novel technique based on the combination of motion compensation (MoCo) and speckle tracking to quantify the 2D motion and tissue velocities of the LV (Joos et al., 2018). The 2D motion estimation was performed using standard cross correlation combined with three different subpixel adjustment techniques. They evaluated the proposed model on in vitro and in vivo in the four-chamber view of 10 volunteers, and their estimated in vitro velocity vectors derived from STE were consistent with the expected values, with normalised errors ranging from 4% to 12% in the radial direction and from 10% to 20% in the cross range direction.

Ouzir et al. (2018) introduced a cardiac motion estimation technique using the sparse properties of motion when decomposed on a dedicated dictionary. They formulated the motion estimation problem as a weighted energy minimisation in an optical flow framework with combined spatial and sparse regularisation. They evaluated the

proposed method on synthetic data, realistic simulation sequences with available GT and two sequences of in vivo images. Their results show the interest of the proposed approach for 2D cardiac ultrasound imaging (Ouzir, Basarab, Lairez et al., 2018).

Some ST methods are based on BM. Since BM can provide a robust estimation of the motion by comparing the similarity between blocks of two images or two video frames, it has been commonly used in the ultrasound field (Tavakoli et al., 2008). During BM, a block of pixels exists in the first frame (known as reference or source frame) will compare with the second frame to search and find this block in the next frame. This description is valid based on the assumption that the reference block remains stable over time. Motion (pixel velocities) is also valid if the frame rate is adequately high. To overcome this limitation and to estimate the velocity of a block in different situations, such as conversion, rotation and scaling (Dufaux and Moscheni, 1995), the BM model is recommended for use.

6.3 Main Contributions

The contributions of this chapter can be listed as follows:

- The feasibility of adopting an optimisation-based BM algorithm to perform speckle tracking in echocardiographic image sequences using was investigated.
- The proposed technique was evaluated using a publicly available synthetic echocardiographic dataset with known GT (i.e., exact values for displacement vectors) from several major ultrasound vendors, and for healthy/ischaemic cases. The results were compared with the results from the classic (standard) 2D BM.

6.4 Synthetic-Dataset

This section describes the employed synthetic cardiac dataset which is publicly available, and the GT is known for the speckle tracking in the 2D echocardiographic images (Alessandrini et al., 2017). This dataset [1] consists of simulated ultrasound

[1] https://gbiomed.kuleuven.be/english/research/50000635/50508167/open-data

images from 7 major vendors such as GE, Hitachi-Aloka, Esaote, Philips, Samsung, Siemens, and Toshiba.

The simulation process is briefly described here, and further details can be found in (Alessandrini et al., 2017). The ultrasound images were simulated from a cloud of point scatterers (scatter map), and using an ultrasound simulator (H. Gao et al., 2009). To take realistic speckle texture for each vendor, scattering amplitude was sampled from a 2D real clinical recording ultrasound as a template. Then, an electromechanical cardiac model was used to relocate the scatterers inside the myocardium and to have a realistic heart motion in the simulated images. Moreover, synthetic probe settings such as scan depth, focus depth, beam density, etc. were specialized by using the values communicated by each vendor upon signature of nondisclosure agreements. They have defined GT displacement and strain in agreement with the recent recommendations from the EACVI/ASE/Industry task force (Voigt et al., 2014; James D Thomas and Badano, 2013).

The GT has been provided as a set of seed points (36 points) along the longitudinal, and 5 points in radial directions. Points were further subdivided into six segments by splitting the endocardial contour into six parts with the same length as shown in Figure 6.1. Synthetic images were provided for normal (healthy) and ischaemic cases for each vendor. Only the A4C views, which is the most commonly used echo view, was included for longitudinal strain calculation.

The total number of frames for vendors GE, Hitachi-Aloka, Esaote, Philips, Samsung, Siemens, and Toshiba, were 54, 72, 54, 50, 65, 56, and 60 respectively. Also, the image size for each vendor is as follows: GE: (479×616), Hitachi-Aloka: (565×811), ESAOTE: (580×682), Philips: (487×619), Samsung: (381×483), Siemens: (617×736), and Toshiba: (489×636). The pixel depth was 8-bit, providing an intensity resolution of 256 gray levels in the synthetic images.

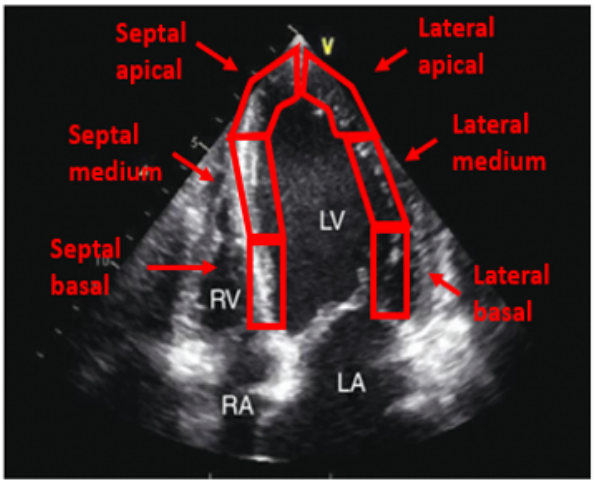


Figure 6.1: an A4C view with the LV myocardium segmentation regions overlaid.

6.5 Method

6.5.1 Standard Block Matching

Classic BM begins by positioning a window on one frame and searching for a pattern with the most similar features within the dimensions of the placed window in the next frame. A cluster of speckles can be combined into one functional unit which is called a kernel; each kernel has a unique fingerprint that is determined using a similarity measure and can be tracked throughout the entire cine loop by the BM algorithm. In the reference frame (first image in Fig 6.2, the current frame or a frame at time t_0) the region of interest (Blue Square) has speckle patterns. In the next frame (a frame at time t+1), a broad region of the image is searched for a similar speckle pattern. The location whose speckle pattern matches best is considered to be the estimated new location of the original kernel, thereby providing an estimated displacement vector.

This procedure is repeated across the whole of the reference frame, obtaining a displacement map between the two images. Repeating this procedure across the whole image sequence produces a vector field across space and time. In this study, Sum of Squared Differences (SSD) is used as a similarity measure which calculates the difference between the intensity pattern of a grid of pixel (original kernel) in one frame and a set of identically sized kernels in the next frame, to find the best-

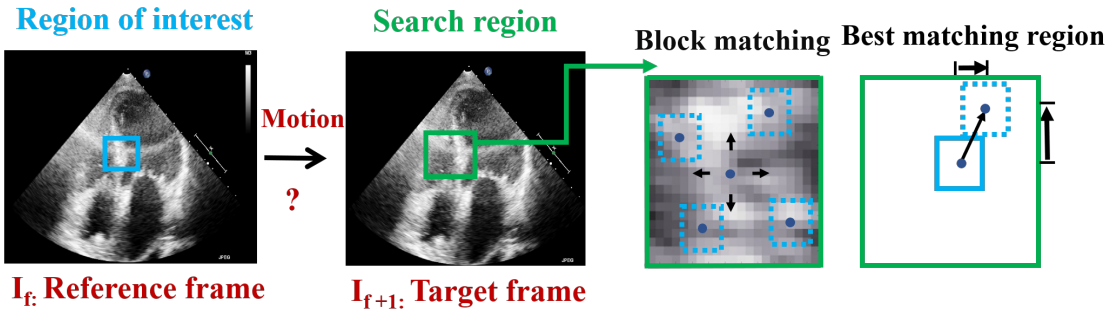


Figure 6.2: Speckle tracking using BM where a region in the image (kernel) is selected and sought for in the next image by sequentially trying out different positions, testing the similarity between the kernel and the pattern observed in that position. The position where the similarity between the kernel and the observed pattern is maximal is accepted as the new position of the original kernel.

matched kernel. Assuming a (m×n) kernel, the comparison between a kernel in the current reference frame (I_1) and a kernel in the target frame (I_2) moved by (p, q) pixel is:

$$SSD(p,q) = \sum_{i=0}^{m}\sum_{j=0}^{n}\left(I_1\left(i,j\right) - I_2\left(i+p, j+q\right)\right)^2 \tag{6.1}$$

where p and q are shift components along the x-direction and y-direction, respectively. The lowest SSD value indicates the most probable direction of the movement of the tissue. Effectively, the BM algorithm tracks the speckles by minimising a cost function. This method is based on the assumption that the SSD value should gradually increase as blocks move further away from the best-matched kernel. Since the smallest step size within the search window is one pixel, it is only able to evaluate the displacement vector to one pixel. To achieve sub-pixel accuracy, a parabolic fitting method was implemented (Gergonne, 1974).

To estimate the motion with sub-pixel precision in the spatial movement, two orthogonal parabolic curves were fitted to the horizontal and vertical of SSD values along with the best matching position. The local minima of the fitted curves were then selected as the final solution, which allows the displacement vector to be evaluated with sub-pixel precision. Based on the parabolic model, denoted by the equation 6.2 where a, b and c are the real constant values, the minimum of the curve can be

found by differentiating and setting the derivative to zero, as shown in the equation 6.3:

$$yy = ax^2 + bx + c \tag{6.2}$$

$$\frac{yy}{dx} = 2ax + b = 0 \qquad x = -\frac{b}{2a} \tag{6.3}$$

Substituting the SSD values for each of these three-data points into the equation 6.2 will give:

$$P_1 = a - b + c \qquad P_2 = c \qquad P_3 = a + b + c \tag{6.4}$$

where P_2 is the minimum SSD value from the kernel, P_1 and P_3 are SSD values from the neighbouring position on either side. The sub-pixel shift x_0 was computed by:

$$x_0 = \frac{P_1 - P_3}{2P_1 - 4P_2 + 2P_3} \tag{6.5}$$

This was done for horizontal and vertical components separately, and the shift values were added to the corresponding integer displacements (p and q in equation 6.1) to obtain sub-pixel accuracy.

6.5.2 Proposed Optimised Block Matching Approach

In this study, a new displacement estimation method is introduced by formulating the tracking as an optimisation problem that jointly penalises intensity disparity and motion discontinuity and is, therefore, more robust to the signal decorrelation when compared to previous approaches. The speckle tracking algorithm combines the BM algorithm with a smoothness constraint for a neighbourhood of kernels, and minimises the following cost function:

$$Costfunction = \sum_r (E_{SSD} + \lambda E_N) \tag{6.6}$$

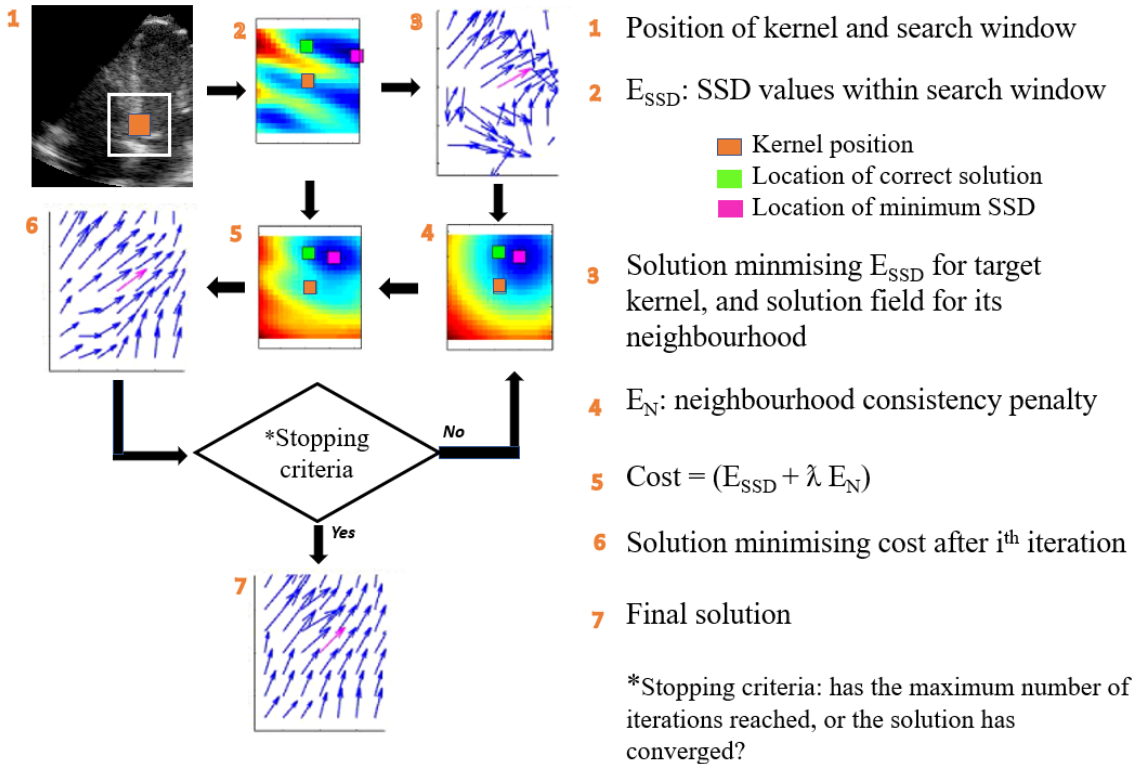


Figure 6.3: Flowchart showing the steps involved in solving the proposed optimisation-based tracking algorithm.

where r is the total number of kernels being tracked, E_{SSD} is the sum of r SSD values.

$$Penalty function = \sum_{r}(\lambda E_N) \tag{6.7}$$

The second component of the cost function ($\sum_r E_n$) which defined in equation 6.7 is a penalty function for speckle (i.e. intensity) decorrelation which penalises the motion discontinuity, and λ is the regularisation weight. This optimisation problem is then solved iteratively.

For the first iteration of the tracking algorithm, the calculated displacement vector field will be smoothed by applying a median filter with kernel size identical to the neighbourhood size. An overall representative displacement vector for the neighbourhood is then obtained by taking the average of all kernels in the neighbourhood. Then, the difference between each potential position in the search window for a kernel and the representative vector is calculated. This is done for r kernels being tracked, and the sum will be the term (∑r En) in equation 6.6. In the next stage,

the overall cost function for each kernel's candidate positions will be computed, incorporating the original SSD values and the penalty term obtained in the previous step. An updated displacement vector field will then be computed by taking each kernel's candidate position with the lowest-cost value, and the process is repeated by estimating a new average representative vector. After each iteration, the new displacement field will be used as an input to the next iteration until either no further changes are observed, indicating the optimisation problem is converged, or a maximum number of iterations is satisfied. Figure 6.3 provides an overall view of the working principles of the proposed tracking algorithm.

6.5.3 Tracking Parameters

The standard BM was carried out with a kernel size of (11×11) pixels with a spacing of 1 pixel, providing a dense solution. This kernel size deemed to be a good compromise for the optimum tracking accuracy.

The size of a search window is also important since a small size would result in the algorithm failing to capture the larger displacements occurring between consecutive frames, and excessively large search window sizes would result in features outside the maximal feasible translation distance to be evaluated as possible links. An optimum size can be estimated from the maximum possible velocity of the myocardium, frame rate, and spatial resolution of the images (i.e., pixels per mm). However, due to the lack of such information being available about the synthetic sequences made accessible, the trial and error method adopted to estimate a reasonable size for the search window, which turned out to be 21×21 pixels (central pixel ±11 pixels). The, this adopted size verified by reviewing the GT (maximum simulated displacement between any pair of frames) across all image sequences.

For the optimised BM approach, the number of iterations was set to 20, which was deemed to be a good compromise between the accuracy and computational run time; a threshold for which the solution was converged, and any further update in the displacement vectors was insignificant. The parameter λ was 0.3, giving more emphasis to the data term versus the regularisation term in equation 6.6. Larger

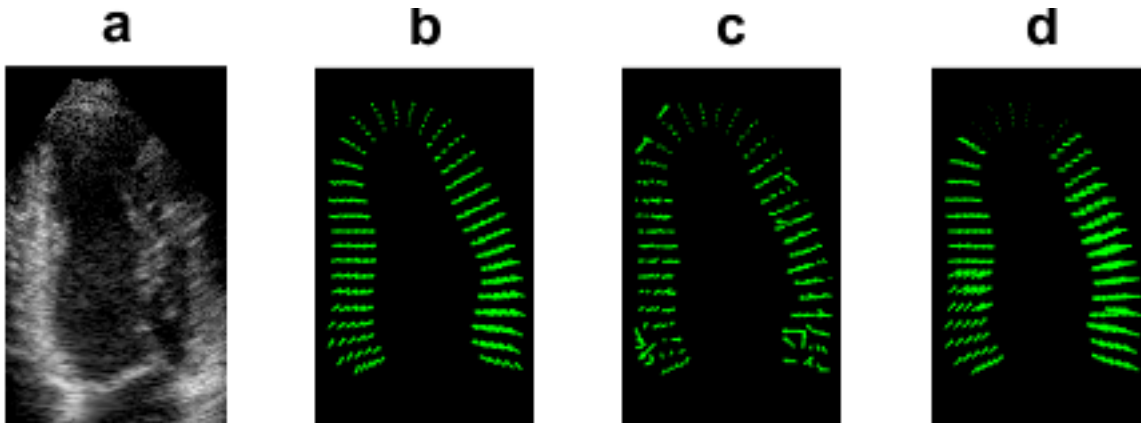


Figure 6.4: An example A4C from the Siemens healthy sequence and corresponding displacement vector fields during the rapid ejection phase (peak systole): (a) zoomed-view of LV cropped from the original image, (b) ground-truth, (c)-(d) displacement fields obtained from standard BM and optimised BM approach in the rapid ejection phase, respectively. Corresponding Figures for other vendors are provided in Appendix A.

values of λ tend to heavily regularise the displacement vectors, which would result in an unrealistically uniform vector field where most of the vectors are aligned. A neighbourhood of (45×45) kernels were included in the iterations for updating the solution for the central kernel. The tracking accuracy was estimated by comparing the displacement field obtained from the speckle tracking algorithms and the GT.

6.6 Evaluation Metrics

In order to evaluate the efficiency of the proposed ST model, displacement and velocity have been measured. Displacement defines the distance that certain speckle features (cardiac structure) have moved between two consecutive frames. Velocities also reproduce displacement per unit of time, that accounting for how fast the location of a speckle feature changes. Since velocity and displacement are vectors, they have direction and amplitude. Therefore, they can be examined through different spatial movements along the anatomic coordinates of the cardiac chambers (longitudinal components), which is especially relevant for the characterization of myocardial mechanics (Mor-Avi et al., 2011; Voigt et al., 2015).

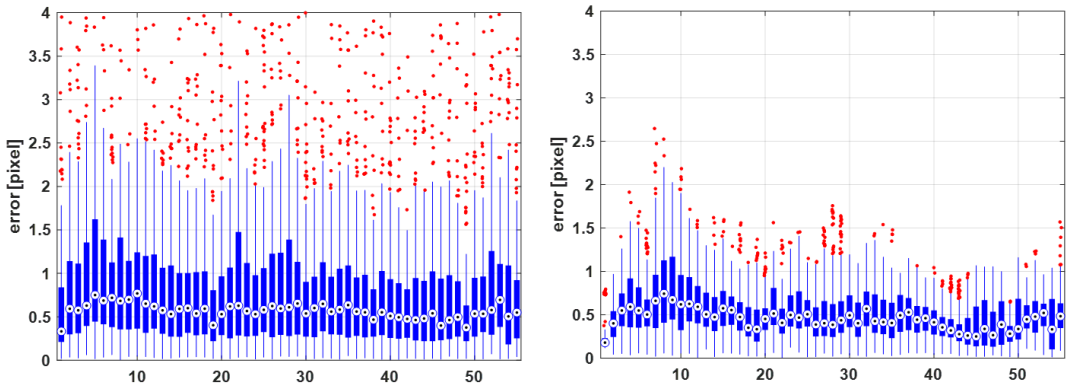


Figure 6.5: Boxplots of the error for the healthy sequence from Siemens. The error is computed as the magnitude of the difference between the calculated and ground-truth displacement vectors, and is provided for standard (left) and proposed (right) tracking methods. The x-axis shows the frame number. The red points represent outliers.

6.7 Experimental Results and Discussion

6.7.1 Displacement Vector Field

The tracking parameters were similar for all vendors and cases. The algorithm returned a dense displacement field between pairs of consecutive frames. Figure 6.4 illustrates an example A4C view from the healthy Siemens sequence in the rapid ejection phase (peak systole), together with the corresponding GT. The computed displacement vector field by the two tracking approaches (standard BM and optimised BM approach) is also shown. The presence of noise in the results is evident in the standard BM technique, whereas the optimised BM approach seems to suffer less from this problem.

Figure 6.5 shows the distribution of error for the same image sequence, obtained from both tracking methods. The displacement errors across all vendors for their corresponding healthy image sequences are shown in Figure 6.6.

As can be seen in Figure 6.5, the optimised BM approach suffers less from the presence of outliers and noise spikes in the computed displacement field. The significant errors in standard BM appear to correspond to the cardiac phases when the heart

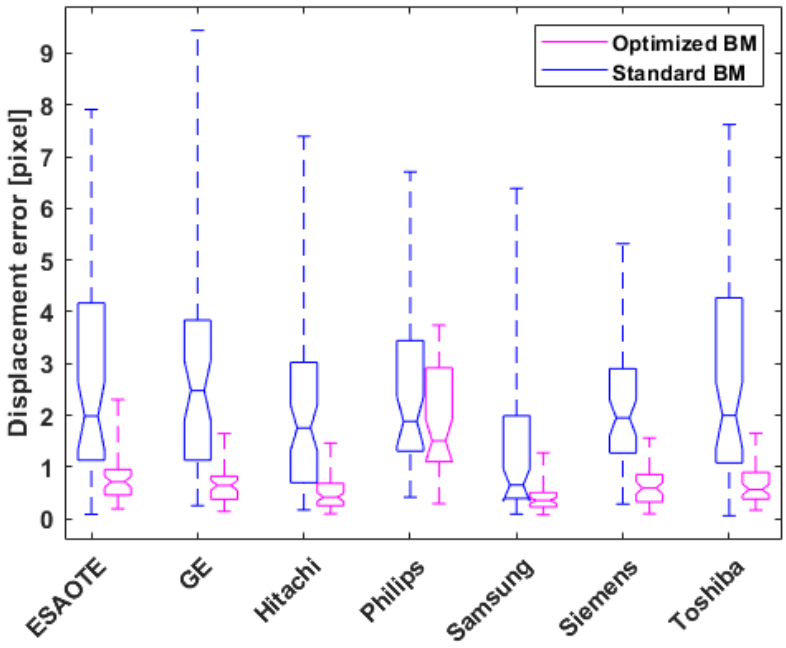


Figure 6.6: Displacement error for the healthy A4C synthetic sequences across all vendors for the two speckle tracking approaches. The horizontal line represents mean; the box signifies the quartiles, and the whiskers represent the 2.5% and 97.5% percentiles.

muscle has the highest velocities, which happen during the rapid ejection phase. For such instances, the magnitude of the displacement is high, and the deformations are relatively large, resulting in lower SSD peaks (or other similarity measures such as correlation-coefficient) in BM. Therefore, secondary peaks caused by random correlations between speckle kernels can sometimes exceed the actual peak. This effect can produce "peak-hopping" artefacts in which a secondary peak is chosen as the best match within a search region, giving rise to significant errors in displacement and deformation estimates. However, the optimised BM approach seems to be less prone to this phenomenon as the fidelity of the solution is checked by the neighbourhood consensus representing the overall motion of the myocardium. The optimised BM approach demonstrates consistently lower errors across all vendors except Philips (Figure 6.6). In the case of synthetic sequences from Philips, the two tracking approaches behave similarly, with the optimised BM approach performing slightly better.

6.8 Execution Time

The proposed speckle tracking algorithms were implemented in C++ programming language. Currently, it takes a maximum of 10s to process a pair of ultrasound frames using an Intel Xeon E5630 CPU, with an internal clock frequency of 2.53 GHz. The focus in this chapter was the accuracy of the tracking results, for offline analysis of the echocardiographic studies. The follow-up studies can look into the implementation of the algorithms on the Graphics Processing Units for parallel computations, from which >1000-fold increase in the processing speed can be expected. This should make the run time feasible for a real-time application.

6.9 Summary

An optimised-based speckle tracking echocardiography algorithm was proposed in this chapter. Its performance was evaluated using a publicly available synthetic echocardiographic dataset with known GT. The results showed improved performance compared with the standard BM in estimating the displacement vector. Next chapter will apply the proposed technique further to calculate the strain values from the tracked displacements.

Chapter 7

Strain Imaging

7.1 Introduction

When cost is immaterial, techniques such as cardiac MRI with semi-automated quantification software improved accuracy and precision, but this is not practical in everyday cardiology. New echocardiographic techniques, such as strain imaging, have emerged as promising quantitative tools in measuring LV function with superior prognostic value to the EF for predicting adverse cardiac events (Kalam, Otahal and Thomas H Marwick, 2014).

Strain imaging have been used to evaluate myocardial function in a wide range of cardiac conditions (D. Y. Leung and Ng, 2010). Strain imaging allows measurement of active tissue deformation in three directions including longitudinal shortening, circumferential shortening, and radial thickening. As the most relevant and clinically useful marker, this thesis has focused on the longitudinal direction. Strain can be obtained from both TDI and 2D B-mode imaging (Dandel and Hetzer, 2009) ultrasound modalities. In the case of B-mode images, speckles are tracked using speckle tracking during the cardiac cycle to provide information of the lengthening or shortening of the myocardial segment. A significant advantage of 2D speckle tracking in comparison with the strain derived from tissue Doppler is that 2D speckle tracking is angle independent and less affected by artifacts (Teske et al., 2007). Therefore, B-mode strain imaging has been widely accepted and perused as the modality of choice for measuring myocardial deformations.

Clinical feasibility of strain resulting from STE has been shown in many studies (Fer-

raiuoli et al., 2019; Rodriguez et al., 2014; Joos et al., 2018; Hui and Xinhua, 2020; D'hooge, Konofagou et al., 2002; D'hooge, Heimdal et al., 2000; D'hooge, Bijnens et al., 2002). For example, strain has been used for the detection of myocardial ischaemia; it may apply after coronary reperfusion to predict infarct size; it is suggested for patients during chemotherapy to detect a decline in cardiac function early. Similarly, strain has been proposed to estimate the risk of ventricular arrhythmias; it may apply to find the optimal position for the pacing lead in the LV free wall in the evaluation of patients after implantation of cardiac resynchronisation therapy (Smiseth et al., 2016).

Although, there has been significant research and advances in strain imaging, use of strain imaging is technically challenging, and have not been successfully incorporated into everyday clinical practice. Continuing technical development and further research are expected to improve the quality of the strain quantification, and more general acceptance of the strain imaging in echocardiography (D. Y. Leung and Ng, 2010).

In this chapter, an overview of strain imaging studies in the literature will be illustrated first. This is followed by evaluation metrics. Finally, the proposed BM model introduced in Chapter 6 will be applied and evaluated on public synthetic datasets to measure strain.

7.2 Previous Work on Strain Imaging

D'hooge et al. (2002) introduced a study on a healthy 29 year old volunteer using VIVID V, GE Vingmed, Horten, Norway, ultrasound scanner. The data were acquired in both the axial and lateral dimensions of the interventricular septum with high temporal resolution and post processed to obtain strain. The direct comparative measurements were not reported while they showed the negative strain during systole. They observe the limitation which the strain in the lateral plane was slightly noisier than in the axial plane (D'hooge, Konofagou et al., 2002).

Ingul et al. (2005) evaluated three methods of strain measurement including a com-

bination of TDI and comparing manual and automated measurements as well. They stated that the automated methods were as effective as the manual method to differentiate between normal and infarcted segments. They found that whilst the manual technique took 11 min to collect the data, the automated technique reduced this to only 2 min. They also demonstrated the angle dependency of Doppler-derived strain and superiority of speckle tracking in measuring strain with greater accuracy (Ingul et al., 2005).

Ouzir et al. (2017) introduced a method based on the sparse representation and dictionary learning to estimate the cardiac motion of 2D echo images. They evaluated the proposed method based on motion estimation accuracy and strain error on one dataset with available GT, including four sequences of highly realistic simulations as well as on both healthy and pathological sequences of in vivo data. The in vivo strain analysis demonstrates that meaningful clinical interpretation can be achieved from the estimated motion vectors (Ouzir, Basarab, Liebgott et al., 2017).

Østvik et al. (2018) developed a U-Net type of CNN to classify muscle tissue, and partitioned into a semantic measurement kernel based on LV length and ventricular orientation. They predicted dense tissue motion using stacked U-Net architectures with image warping of intermediate flow to tackle variable displacements. They used a mixture of real and synthetic data for training. The resulting segmentation and motion estimates was fused in a Kalman filter and used as basis for measuring Global Longitudinal Strain (GLS). The qualitative assessment showed comparable deformation trends as the clinical analysis software. They have reported the average deviation for the GLS -0.6 $\pm$ 1.6% for A4C view (Østvik, Smistad, Espeland et al., 2018). The same group was developed a motion estimator based on a PWC-Net architecture, which achieved an average end point error of (0.06 $\pm$ 0.04) mm per frame using simulated data from an open access database (Ostvik et al., 2021).

Joos et al.(2018) proposed a novel technique using the combination of speckle tracking and motion compensation (MoCo) on high-frame-rate B-mode images to quantify the tissue velocities of the LV. They evaluated their method on in vitro and in vivo in the A4C view of 10 volunteers. They obtained GLS of the LV from speckle track-

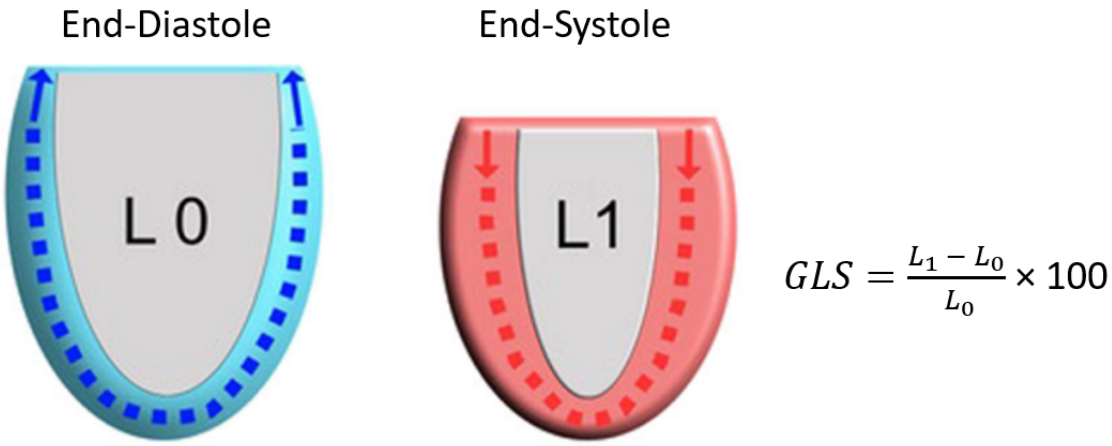


Figure 7.1: Illustration of GLS. L, length; L_0, total longitudinal length of the LV border in diastole; L_1, total longitudinal length of the LV border in systole.

ing in 10 subjects and compared to the results provided by a clinical scanner which achieved p value of 0.33 (Joos et al., 2018).

7.3 Strain Calculations and Evaluation Metrics

In order to evaluate the efficiency of the proposed ST model presented in Chapter 6, GLS and regional/segmental strain have been measured. Strain describes the deformation of an object normalised to its original shape and size. Using the displacement vectors obtained from the speckle tracking, and according to the recent recommendations from the EACVI/ASE/Industry task force (Voigt et al., 2014; James D Thomas and Badano, 2013), strain can be calculated as:

$$\epsilon\,(t) = \frac{L(t) - L_0}{L_0} \tag{7.1}$$

where $L(t)$ is either the length of a segment (in case of segmental/regional strain) or the length of the LV contour (in case of GLS) at a given point in time, and L_0 is the reference length at the reference time t_0. In the case of computing GLS, L_0 is the total longitudinal length of the LV border in ED frame. Strain is a dimensionless entity, reported as a fraction or percent (Dandel, Lehmkuhl et al., 2009; Sanfilippo et

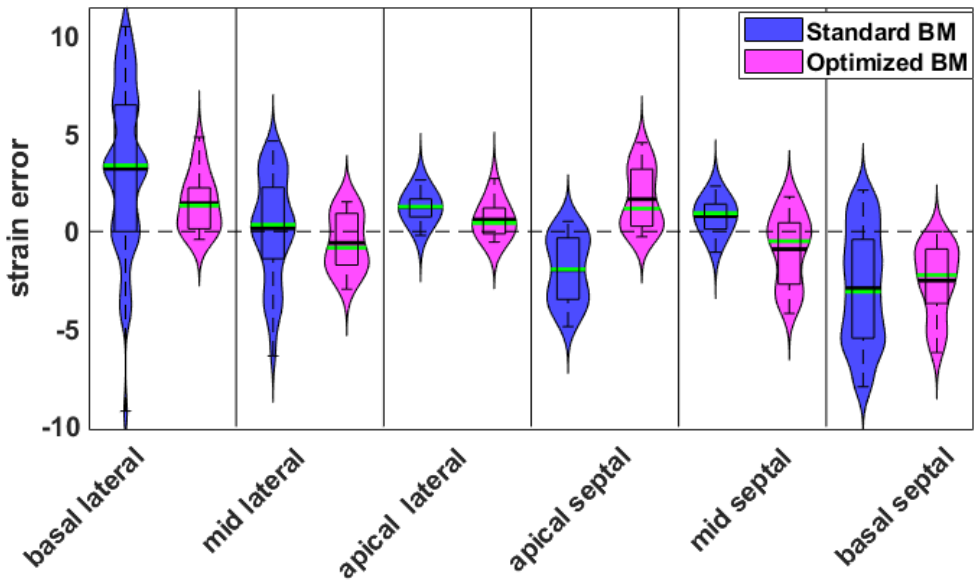


Figure 7.2: violin plots of the error in the segmental strain measurements for the healthy synthetic sequence from Siemens. The solid black line represents mean, and the green line represents the median; the box signifies the quartiles, and the whiskers represent the 2.5% and 97.5% percentiles.

al., 2018). The conceptual assessment of myocardial function with GLS is illustrated in Figure 7.1. [1]

7.4 Experimental Results

Since this thesis has focused on strain measurements in the LV only, the images were cropped manually before speckle tracking process, by considering a rectangle containing the LV. However, the initial positioning of the tracking kernels was automatic.

Regional (Segmental) longitudinal strain values were calculated from the estimated displacement vector field which explained in Chapter 6. Figure 7.2 displays the violin plots of the regional strain error (the difference between the speckle tracked and the GT) for all LV segments.

GLS values were computed from the estimated displacement vector fields for the

[1] As per our discussions with the project clinical partners at Imperial College London, GLS measurements in LV was assumed to be the most clinically relevant and useful component.

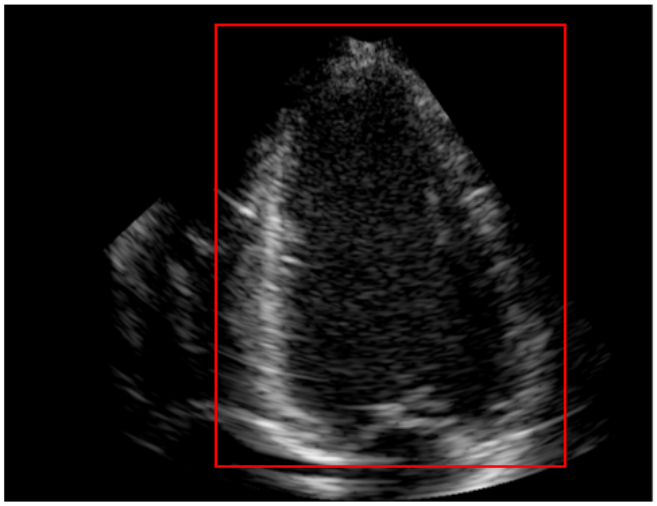


Figure 7.3: Example of synthetic image (Toshiba vendor). The rectangle displays the region of interest considered for speckle tracking.

healthy and ischemic sequences across all vendors. The results are provided in Figure 7.4 and 7.5 for the standard BM and the optimised BM approaches, respectively. An improvement in the case of the optimised BM approach is evident. Figure 7.5 illustrates that by minimising the cost function (Chapter 6, equation 6.6), the estimated strain values are more reliable for both healthy and ischaemic cases and across all vendor except the healthy case for the Samsung image sequences, where a considerable bias between the calculated strain and the GT is evident. This is most likely due to the missing/smeared walls in the images as shown in Appendix A, where the algorithms fail to return meaningful speckle tracked displacement vectors.

In the case of synthetic sequences from Philips, the two tracking approaches behave similarly, with the optimised BM approach performing slightly better. Similar behaviour is observed in the calculated strain measurements. A considerable improvement in the basal segments (both lateral and septal) can be seen in the optimised BM approach when compared with the standard BM approach. This is likely to be because of the fast-moving heart muscles in these segments for which the standard BM struggles to track, most likely due to the peak-hopping artefacts. For the apical segments, where the site of measurement is in the vicinity of the apex and, therefore, moves at relatively lower velocities, the error in both methods drops, relative to the corresponding basal segments.

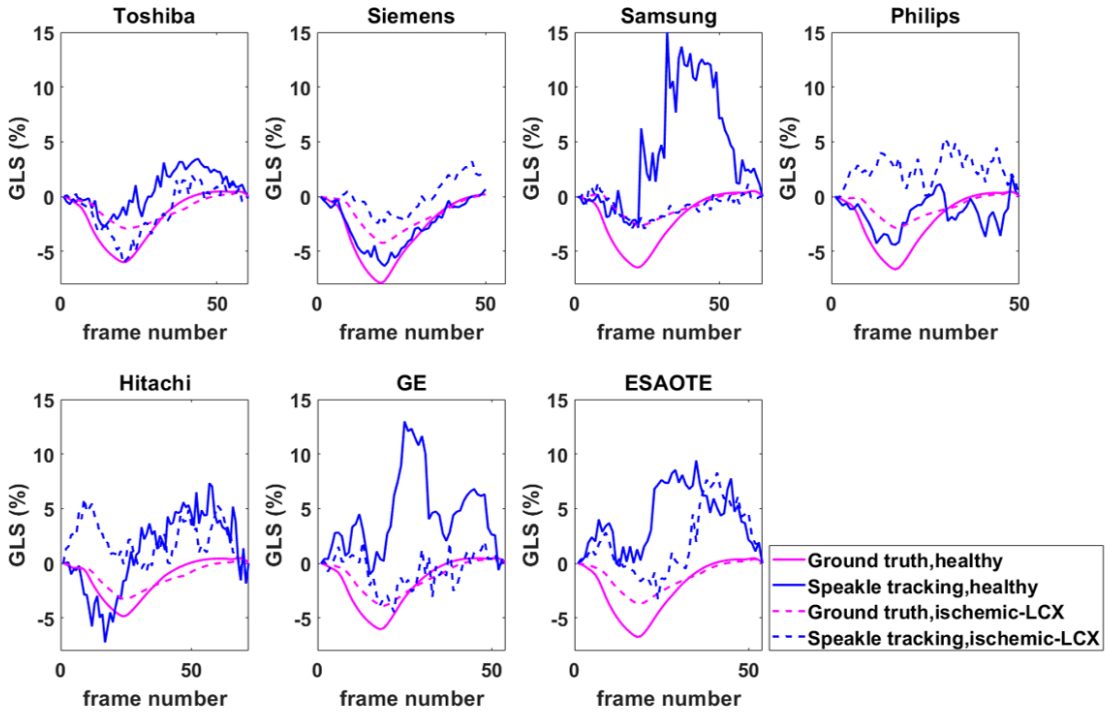


Figure 7.4: Comparison of GLS measurements obtained from the standard BM approach for the healthy and ischemic-LCX (Ischemic-left circumflex coronary artery) cases across all vendors with the known ground-truth. The solid and dashed blue lines represent the calculated strain values for healthy and ischemic cases, respectively. The solid and dashed magenta lines indicate the corresponding ground-truth.

The statistical analysis of standard and optimised BM approaches has been presented for the GLS measurements for the healthy and ischaemic cases across all vendors in Tables 7.1 and 7.2, respectively. As shown in Table 7.1 and 7.2, overall, the optimised BM approach demonstrated better performance in estimating the GLS values in comparison with the standard BM. In case of ischemic GE sequences, a close to zero correlation coefficient for the standard BM indicated very poor tracking results, where the optimised approach seems to be offering more reliable results, with a correlation coefficient of 0.98. For the ischemic sequences from Philips, however, both tracking approaches suffer from poor strain measurement errors, with correlation coefficients of 0.32 and 0.34, respectively. The simulated image sequences for both vendors have relatively poorer image qualities, with segments of the myocardium is missing/invisible in the simulated imaging plane, where the tracking algorithms struggle to follow the speckle movements between consecutive images. For all other vendors, the optimised BM approach demonstrates an acceptable level of accuracy.

Table 7.1: Statistical analysis of standard and optimised BM approaches for the GLS measurements for healthy sequences across all vendors; the slope of the regression line (α), correlation coefficient (ρ), bias (μ), upper limits of agreement (ULOA), and lower limit of agreement (LLOA) are provided.

	Standard BM					Optimised BM				
Vendor	α	ρ	μ	ULOA	LLOA	α	ρ	μ	ULOA	LLOA
Hitachi	0.28	0.57	2.34	8.05	-3.36	0.86	0.93	0.29	1.61	-1.03
Toshiba	0.93	0.72	2.42	5.48	-0.63	1.05	0.98	0.20	0.91	-0.50
Esaote	0.20	0.23	6.17	12.62	-0.27	1.08	0.99	-0.17	0.51	-0.86
Samsung	0.14	0.33	6.15	16.02	-3.71	1.09	0.89	1.20	3.34	-0.93
Siemens	1.27	0.97	0.40	2.02	-1.21	1.18	0.99	0.48	1.44	-0.46
Philips	0.81	0.51	0.55	4.63	-3.53	1.21	0.99	0.19	1.12	-0.72
GE	0.04	0.08	5.96	14.10	-2.16	1.07	0.99	0.05	0.61	-0.49

Table 7.2: As Table 7.1, but for ischaemic sequences.

	Standard BM					Optimised BM				
Vendor	α	ρ	μ	ULOA	LLOA	α	ρ	μ	ULOA	LLOA
Hitachi	0.06	0.18	1.82	10.20	-6.55	0.69	0.74	-0.12	1.96	-2.21
Toshiba	0.28	0.49	1.74	6.34	-2.84	0.84	0.97	-0.26	0.64	-1.17
Esaote	0.18	0.28	3.77	8.96	-1.42	1.07	0.97	0.35	1.15	-0.43
Samsung	0.26	0.43	0.63	5.13	-3.87	0.90	0.86	0.54	2.07	-0.99
Siemens	0.73	0.66	0.74	3.38	-1.88	1.08	0.95	0.47	1.56	-0.60
Philips	0.02	0.32	0.48	40.75	-39.79	0.02	0.34	-1.89	31.98	-35.77
GE	-0.07	-0.14	2.15	9.15	-4.85	1.05	0.98	0.29	0.89	-0.30

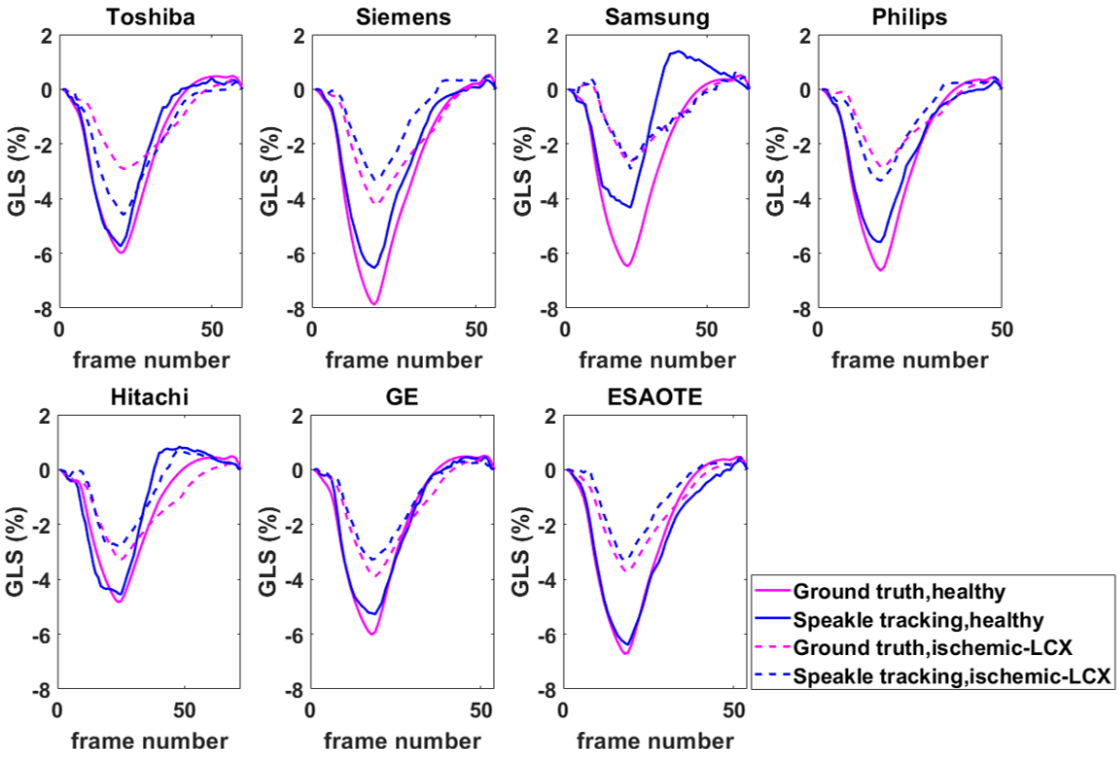


Figure 7.5: Same as Fig 7.4, but for the optimised BM approach.

Considerable biases are observed in both healthy and ischemic cases in GLS values obtained from the standard BM approach and for some of the vendors (Figure 7.4).

7.5 Discussion

It is worth noting that such poor results are less likely to be observed when using vendors' software packages. This is because here the results have presented from a purely BM step where no additional post-processing is applied. From a clinical image sequence, speckle-matching alone never provides an unambiguous, obviously correct, velocity field. Physical limitations of ultrasound, and out-of-plane motion, prevent perfect speckle matching. There are often several ways that a speckle pattern in one frame could transform into its counterpart in the next frame. Therefore, this study presume the current strategy undertaken by most vendors is a 2-step process. First, calculate the displacement vector field maximising the match between successive frames (i.e. standard BM). Second, apply automated "common sense" editing that weeds out implausible vectors, and instead infer values using regions adjacent in space and/or time (i.e., spatial or temporal filtering).

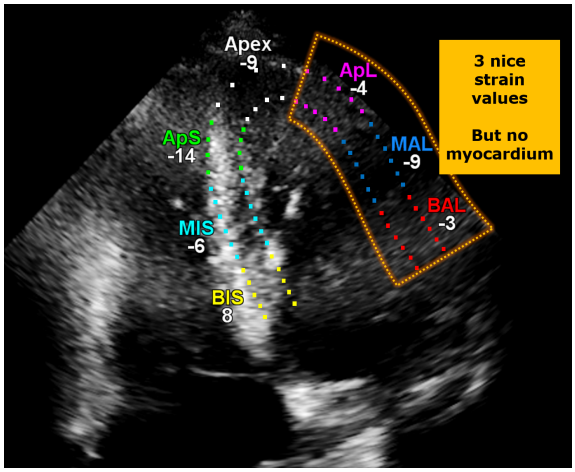


Figure 7.6: Example of a (presumably) “common sense” editing on one frame, where 3 regional strain values are given for the lateral wall, but there is no visible myocardium to be tracked in that region (QLAB 10.0, Philips).

Figure 7.6 illustrates an example providing evidence for this "common sense" editing, where 3 reasonable strain values are given for the lateral wall, but there is no myocardium to be tracked in that region. Exactly how this "common sense" work has a large effect on all downstream results including strain. This can potentially explain the persistent contradiction between vendors despite their standardising definitions for the acquisition and nomenclature. Task Force acknowledging the inability to resolve the vendor discrepancy, and recommending follow-up measurements to be done with the same software as before. This causes logistical problems (if a hospital has >1 vendor) or vendor lock-in (Lang, Badano et al., 2015).

Interestingly, a recent study (Negishi et al., 2013) has concluded that post-processing is the most important determinant in inter-vendor variation, with differences in acquisition having a small effect. None of the vendors included in this study has disclosed its algorithms for strain measurements. Therefore, could not reproduce the result of their corresponding software packages for a direct comparison here.

Speckle decorrelation is signal- and motion-dependent. Therefore, it cannot be compensated by simple post-tracking spatial or temporal smoothing. Thus, the proposed approach simultaneously maximises match and penalises implausibility (fusing BM and biological constraints), optimised by minimising the two-element cost function. The optimisation process jointly maximises signal correlation and motion continuity,

eliminating the need for subsequent editing of the raw displacement vectors which is probably the underlying cause of vendor discrepancy.

7.6 Summary

The proposed optimised-based speckle tracking echocardiography algorithm was used to calculate longitudinal strain in echocardiographic images. The results showed improved performance compared with the standard BM in estimating the strain measurements. The proposed tracking method does not require any post-processing or filtering steps and can potentially reduce the variability in strain measurements caused by various implementations of such filtering techniques.

Chapter 8

Conclusions and Future Work

8.1 Conclusion

This thesis focused on contributing to the automation of image interpretation in echocardiography as the most commonly used non-invasive modality, in clinical practice and research, to evaluate the structure and function of the heart. Such automated systems can provide a significant contribution to the clinical procedures by providing clinicians with the assistive decision-making tools to reliably diagnose and treat cardiovascular disease.

The clinical background of cardiology, the significance of echocardiograms, the different varieties of echocardiograms, and ultrasound modalities with a focus on the TTE and B-Mode modality, were provided in **Chapter 2**. Additionally, speckle tracking concept and myocardial deformation parameters were explained.

Chapter 3 provided the technical background on methods used in this thesis such as an overview of neural networks, approaches to neural network design, most common classification, and segmentation architectures. Moreover, an overview of the NAS including search space, search strategy, and performance estimation strategy has been discussed.

In **Chapter 4**, the differentiable architecture search approach was utilised to design a neural network for the automated identification of 14 different anatomical echocardiographic views in a large dataset. The main aim of the model was to design a small neural network architecture for rapid inference while maintaining high ac-

curacy. The impact of 4 different image sizes such as 32×32, 64×64, 96×96, and 128×128 pixels were investigated. The direct correlation between the image quality and classification accuracy was observed.

The influence of different training dataset sizes (i.e. 100%, 50%, 16%, 8%) on the accuracy of the models also was examined. The adopted models derived from DARTS solution appear to be relatively less profoundly affected by the size of training dataset, where both 1-cell-DARTS and 2-cell-DARTS models demonstrate no more than an 8% drop in their prediction accuracy when deprived of the full training dataset. When using 50% of the training dataset, both DARTS-based models exhibit better performance over the deeper networks.

The impact of image quality on the efficacy of the models was also investigated. There is a correlation between classification accuracy and image quality (p-value 0.01). The images labelled as "excellent" quality showed the highest classification accuracy of about 100%. The discrepancy between the model's prediction and the expert annotation is higher in poor quality images.

The model was evaluated on a private dataset of 14 different echocardiographic views and the results were compared with the standard classification CNN architectures. In contrast to the deeper classification architectures, the proposed model has a significantly lower number of trainable parameters (up to 99.9% reduction), achieved comparable classification performance (accuracy 88.4-96.0%, precision 87.8-95.2%, recall 87.1-95.1%) and real-time performance with inference time per image of 3.6-12.6ms.

Chapter 5 utilised a NAS technique to design a neural network for the automated segmentation of LV in 2D echocardiographic images. Three different datasets of echocardiographic images including one public and two private datasets were used for training and testing the proposed models. The proposed model was applied to one public dataset and two private datasets for A4C and A2C of echocardiographic views, and its performance was compared with the state-of-the-art dense prediction architectures such as U-Net, U-Net ++, SegNet, and DeepLabV3ResNet101.

The results revealed that the proposed model outperforms other models for all data-

set. The proposed model on public CAMUS dataset achieved an average DC of 0.944 $\pm$ 0.038, 0.892 $\pm$ 0.042, and 0.919 $\pm$ 0.075 for LV-Endo, LV-Epi, and LA respectively.

The influence of different training dataset sizes (i.e. 100%, 50%, 16%, 8%) on the performance of the models also was examined. The proposed model is affected relatively less by the size of training dataset. The impact of image quality on the performance of the models was also investigated. Also, the impact of mono and multi-structure learning approaches on the performance of the proposed network was examined.

Chapter 6 presented a novel optimisation-based BM algorithm to perform speckle tracking iteratively. The proposed model was evaluated using a publicly available synthetic echocardiographic dataset with known ground-truth from several major vendors, and for healthy and ischemic cases. The new method of improving resistance to image noise is introduced by applying a penalty for spatial inhomogeneity of velocity to perform speckle tracking iteratively in cardiac synthetic ultrasound image sequences.

The results were compared with the results from the classic (standard) 2D BM. The proposed method presented an average displacement error of 0.57 pixels, while classic BM provided an average error of 1.15 pixels.

In **Chapter 7**, the novel model presented in Chapter 6 was applied to public synthetic images to estimate the segmental and regional longitudinal strain in healthy cases. The proposed method, with an average strain error of 0.32$\pm$0.53, outperformed the classic counterpart, with an average 3.43$\pm$2.84. A similar superior performance was observed in ischaemic cases.

The proposed method does not require any additional ad-hoc filtering process. Therefore, can potentially help to reduce the variability in the strain measurements caused by various post-processing techniques applied by different implementations of speckle tracking.

8.2 Future Work

The proposed methods for automatic classification, segmentation, speckle tracking, and strain calculation were evaluated on different private and public datasets. The evaluation results presented reasonable performances compared to the state-of-the-art results. The potential future directions are summarised below.

8.2.1 Echo View Classification

This study has focused on the rapid and accurate classification of individual frames from an echo cine loop. Such a task will be crucial for a real-time view detection system in clinical scenarios where images need to be processed while they are acquired from the patient and/or where the system is to be used for operator guidance. However, for offline studies and when the entire cine loop is available, classification of the echo videos could also be of practical use.

Some studies have attempted video classification using the majority vote on some or all frames from a given video (Østvik, Smistad, Aase et al., 2019; Madani, Ong et al., 2018). However, this approach does not use the temporal information available in the cine loop, such as the movement of structures during the cardiac cycle. Therefore, a future study could look into using all available information for view detection.

This study investigated 2D echocardiography as a clinically relevant modality. Currently, 3D echocardiography suffers from a considerable reduction in frame rate and image quality, and this has limited its adoption into routine practice over the past decade (Cheng et al., 2018). When such issues are resolved, automatic processing of the 3D modality could also be explored. In the meantime, 2D echocardiography remains unrivalled, particularly when high frame rates are needed.

Also, this study investigated the impact of image quality on the classification accuracy for A4C views only. A more comprehensive examination of the image quality and its influence on the detection of different echo views would be informative.

The dataset used in this study was comprised of images acquired using ultrasound equipment from GE and Philips manufacturers. Although the proposed models do

not make any *a priori* assumptions on data obtained from specific vendors and therefore should be vendor-neutral, echo studies using more diverse ultrasound equipment should still be explored.

Similar to all previous studies, in this study, the dataset originated from one medical centre, i.e. Imperial College Healthcare NHS Trust's echocardiogram database. Representative multi-centre patient data will be essential for ensuring that the developed models will scale up well to other sites and environments.

In this study interpreting the results of the proposed models alongside other proposed architectures in the literature (with a wide range of reported accuracies) was not feasible. This is due to the fact that a direct comparison of the classification accuracy would require access to the same patient dataset. At present, no echocardiography dataset and corresponding annotations for view detection are publicly available.

Moreover, this study examined the proposed neural network using only one manual expert annotations. Future study can consider examining the performance of the models using more than one manual expert judgment to study the discrepancies of the variability of annotations.

However, generally, training a neural network requires large amounts of annotated data, which is often very difficult to obtain, especially in the medical field. To alleviate this problem and to reduce the cost of annotation, an important direction that needs to be study is examining the possibility of self supervised learning to classify echo view images. Recently, self-supervision hold the potential to yield significant improvements in the learning process of the target task (Danu, Ciuşdel and Itu, 2020).

8.2.2 Left Ventricle Segmentation

This study examined the proposed network using manual expert annotations. Due to the label scarcity and high-cost of annotation, future study can consider examining the performance of self supervised or semi-supervised learning for the segmentation of echo images to leverage the large quantity of unlabeled echo images.

Cardiologists usually examine multi-view echocardiographic images in clinical decision-making (Madani, Arnaout et al., 2018). The A2C, A3C, and A4C views are the most commonly used views for the LV functional assessment. This study used A4C for private datasets and A4C and A2C views in public dataset (CAMUS) for the segmentation of echo images. Future study can consider to search and train the proposed model to achieve a network for multi-view such as A2C, A3C, and A4C for echocardiographic segmentation.

Also, this study considered the single frames to train the neural network, however, future study can focus on the sequences which carries the LV information from previous frames to following frames that can help the matching between consecutive frames naturally (Li, W. Zhang et al., 2019).

8.2.3 Speckle Tracking and Strain Imaging

This study considered only the A4C view, which is the most common apical probe orientation. However, no view-specific assumptions were made during the algorithm developments, and the proposed tracking method should, in principle, apply to other echo views. Therefore, future studies would include other standard echocardiographic views such as 2-chamber and 3-chamber.

Additionally, in this study, synthetic image sequences were used for evaluating the performance of the tracking algorithms. This provided the advantage of knowing the exact solution (GT) for the speckle tracking which could be used for error calculations. However, future studies would consider using large datastes acquired echocardiographic image sequences, representing real-world clinical data.

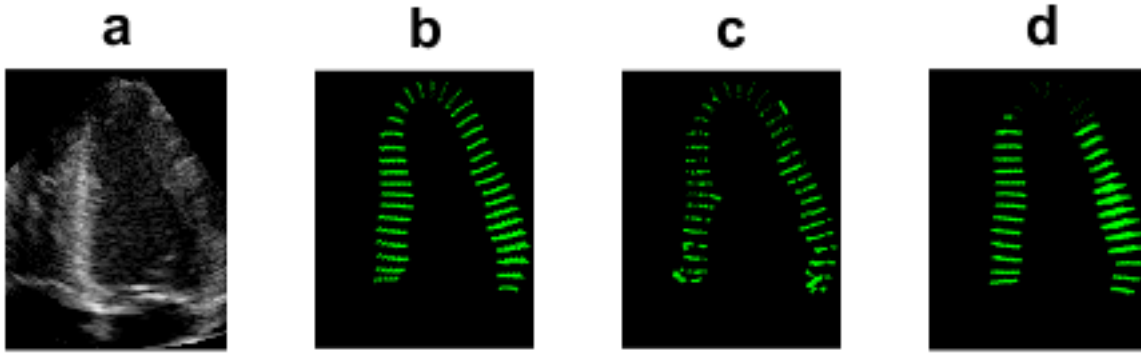
The purpose of this study was to examine the performance of an improved speckle tracking technique in estimating the displacement of strain measurements. Hopefully, this would serve as a stepping stone to addressing the issue of inter-vendor variability, which has become the main limitation to the implementation of this technique in clinical settings. Assuming the vendor discrepancy is partly due to different ”common sense” editing and filtering techniques applied by the vendors to the erroneous speckle tracking results (to make the results see more reasonable), this improved version of

tracking could potentially help in reducing the variability by eliminating the need for all subsequent editing of the results.

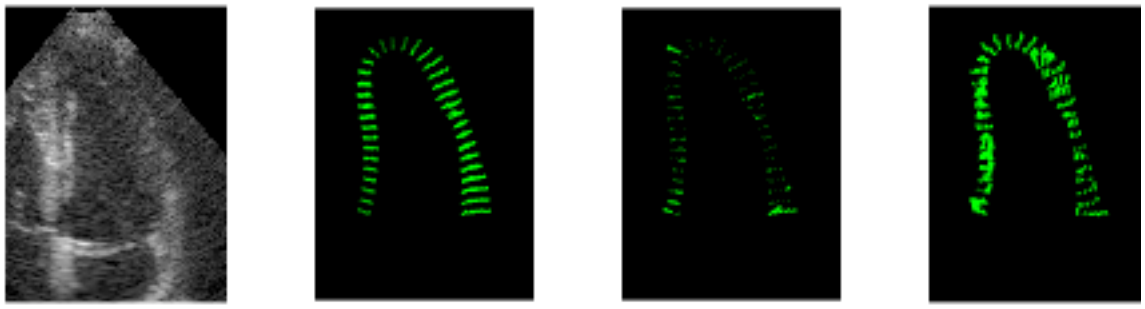
A thorough investigation of this issue would require the use of echocardiograms obtained from the same patient, but using different vendors. The synthetic available and used dataset in this study provides sequences from different vendors and patients. Therefore, a direct comparison of the results to examine the inter-vendor variability was not possible in the current study. A future comprehensive study must examine the potential influence of the proposed tracking algorithm on the inter-vendor variability in the strain measurements.

Appendix A

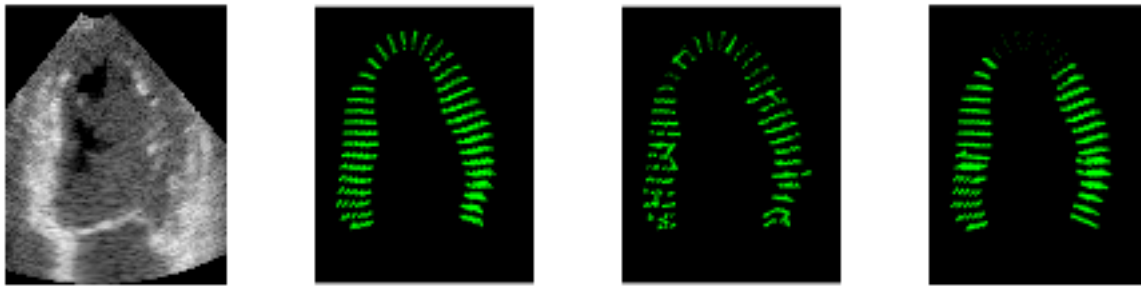
An example A4C from 6 vendors with the healthy sequence and corresponding displacement vector fields: (a) zoomed-view of LV cropped from the original image, (b) ground-truth, (c)-(d) displacement fields obtained from standard BM and optimized BM approach in the rapid ejection phase, respectively.



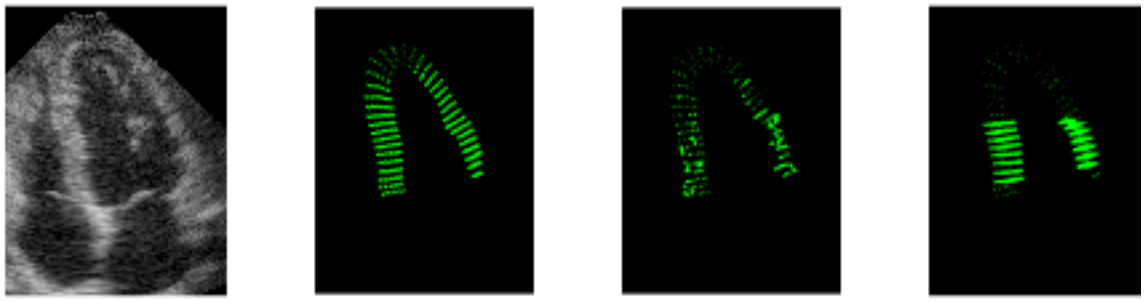
Toshiba



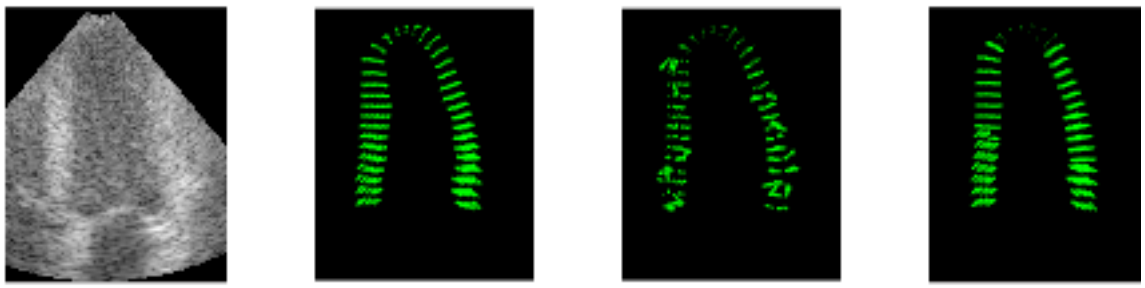
Samsung



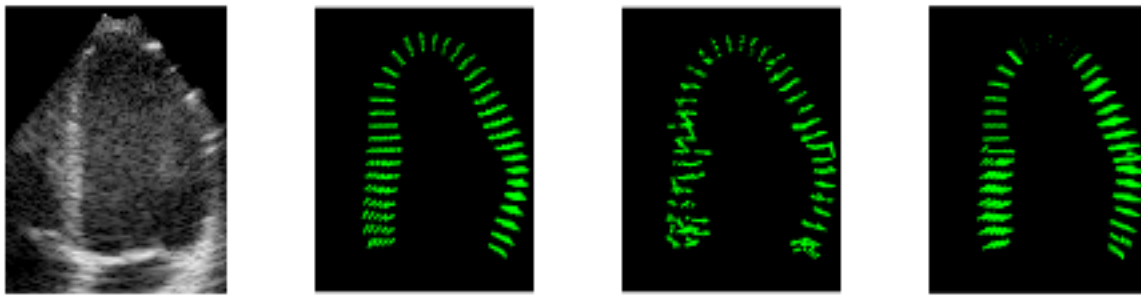
Philips



Hitachi



GE



ESAOTE

Appendix B

List of Publications

- Azarmehr, N., Ye, X., Howes, J.D., Docking, B., Howard, J.P., Francis, D.P. and Zolgharni, M., 2020. An optimisation-based iterative approach for speckle tracking echocardiography. Medical & Biological Engineering & Computing, pp.1-15.
- Azarmehr, N., Ye, X., Janan, F., Howard, J.P., Francis, D.P. and Zolgharni, M., 2019, April. Automated Segmentation of Left Ventricle in 2D echocardiography using deep learning. In International Conference on Medical Imaging with Deep Learning–Extended Abstract Track.
- Azarmehr, N., Ye, X., Sacchi, S., Howard, J.P., Francis, D.P. and Zolgharni, M., 2019, July. Segmentation of Left Ventricle in 2D echocardiography using deep learning. In Annual Conference on Medical Image Understanding and Analysis (pp. 497-504). Springer, Cham.
- Azarmehr N, Ye X, Howard P, Lane E, Labs R, Shun-shin M, Cole G, Bidaut L, Francis D, and Zolgharni M, (2020) Neural Architecture Search of Echocardiographyview Classifiers, Journal of Medical Imaging.
- Lane E S., Azarmehr N., Jevsikov, J., Howard J. P., Shun-shin M. J., Cole G D., Francis D.P., and Zolgharni M., 2021. Multibeat Echocardiographic Phase Detection Using Deep Neural Networks, Computers in Biology and Medicine.
- Labs, R. B., Vrettos, A., Azarmehr, N., Howard, J.P., Shun-shin, M. J., Francis, D.P. and Zolgharni, M., 2019, July. Automated Assessment of Image Quality in

2D Echocardiography Using Deep Learning In ICRMIRO 2020: International Conference on Radiology, Medical Imaging and Radiation Oncology.

– segmentation paper, to be submitted

References

Abduch, Maria Cristina Donadio et al. (2014). 'Cardiac mechanics evaluated by speckle tracking echocardiography'. In: *Arquivos brasileiros de cardiologia* 102.4, pp. 403–412 (cit. on p. 22).

Agarwal, Dhruv, KS Shriram and Navneeth Subramanian (2013). 'Automatic view classification of echocardiograms using histogram of oriented gradients'. In: *2013 IEEE 10th International Symposium on Biomedical Imaging*. IEEE, pp. 1368–1371 (cit. on p. 52).

Albinsson, John et al. (2018). 'Iterative 2D tissue motion tracking in ultrafast ultrasound Imaging'. In: *Applied Sciences* 8.5, p. 662 (cit. on p. 109).

Alessandrini, Martino et al. (2017). 'Realistic vendor-specific synthetic ultrasound data for quality assurance of 2-d speckle tracking echocardiography: Simulation pipeline and open access database'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 65.3, pp. 411–422 (cit. on pp. 111, 112).

Alsharqi, Maryam et al. (2018). 'Artificial intelligence and echocardiography'. In: *Echo research and practice* 5.4, R115–R125 (cit. on p. 109).

Amari, Shunichi et al. (2003). *The handbook of brain theory and neural networks*. MIT press (cit. on p. 27).

Amundsen, Brage Høyem (2015). *It is all about timing!* (Cit. on p. 5).

Anandalingam, G and Terry L Friesz (1992). 'Hierarchical optimization: An introduction'. In: *Annals of Operations Research* 34.1, pp. 1–11 (cit. on p. 60).

Anwar, Syed Muhammad et al. (2018). 'Medical image analysis using convolutional neural networks: a review'. In: *Journal of medical systems* 42.11, p. 226 (cit. on p. 80).

Atzeni, Fabiola et al. (2017). *The Heart in Systemic Autoimmune Diseases*. Elsevier (cit. on p. 21).

Azarmehr, Neda et al. (2019). 'Automated Segmentation of Left Ventricle in 2D echocardiography using deep learning'. In: *International Conference on Medical Imaging with Deep Learning – Extended Abstract Track*. London, United Kingdom. URL: `https://openreview.net/forum?id=Sye8klvmcN` (cit. on p. 82).

Badrinarayanan, Vijay, Alex Kendall and Roberto Cipolla (2017). 'Segnet: A deep convolutional encoder-decoder architecture for image segmentation'. In: *IEEE trans-*

actions on pattern analysis and machine intelligence 39.12, pp. 2481–2495 (cit. on pp. 38, 39, 79).

Bae, Woong et al. (2019). 'Resource Optimized Neural Architecture Search for 3D Medical Image Segmentation'. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 228–236 (cit. on p. 49).

Bahreini Toosi, MH, H Zarghani, H Poorzand et al. (2019). 'Sex-related Left Ventricle Rotational and Torsional Mechanics by Block Matching Algorithm'. In: *Journal of biomedical physics & engineering* 9.5, p. 541 (cit. on p. 7).

Baker, Bowen et al. (2017). 'Designing Neural Network Architectures using Reinforcement Learning'. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. URL: `https://openreview.net/forum?id=S1c2cvqee` (cit. on pp. 40, 42).

Balaji, GN, TS Subashini and N Chidambaram (2015). 'Automatic classification of cardiac views in echocardiogram using histogram and statistical features'. In: *Procedia Computer Science* 46, pp. 1569–1576 (cit. on p. 3).

Bansal, Manish and Ravi R Kasliwal (2013). 'How do I do it? Speckle-tracking echocardiography'. In: *Indian heart journal* 65.1, p. 117 (cit. on p. 22).

Barbosa, Daniel et al. (2014). 'Fast tracking of the left ventricle using global anatomical affine optical flow and local recursive block matching'. In: *MIDAS J* 10 (cit. on pp. 8, 110).

Bello, Irwan et al. (2017). 'Neural optimizer search with reinforcement learning'. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, pp. 459–468 (cit. on p. 32).

Bergstra, James, Daniel Yamins and David Cox (2013). 'Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures'. In: *International conference on machine learning*, pp. 115–123 (cit. on p. 44).

Beymer, David, Tanveer Syeda-Mahmood and Fei Wang (2008). 'Exploiting spatio-temporal information for view recognition in cardiac echo videos'. In: *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, pp. 1–8 (cit. on p. 52).

Blessberger, Hermann and Thomas Binder (2010). 'Two dimensional speckle tracking echocardiography: basic principles'. In: *Heart* 96.9, pp. 716–722 (cit. on p. 22).

Blinn, Justin A, Vitaly Margulis and Ravi V Joshi (2019). 'Transesophageal Echocardiography Imaging of the Inferior Vena Cava and Hepatic Vein Masses'. In: *A&A Practice* 12.8, pp. 295–297 (cit. on p. 19).

Bosch, Johan G et al. (2002). 'Automatic segmentation of echocardiographic sequences by active appearance motion models'. In: *IEEE transactions on medical imaging* 21.11, pp. 1374–1383 (cit. on p. 79).

Bulwer, BE, SK Shernan and JD Thomas (2011). 'Physics of echocardiography'. In: *Comprehensive textbook of perioperative transesophageal echocardiography* 2 (cit. on p. 18).

Cai, Han, Ligeng Zhu and Song Han (2019). 'ProxylessNAS: Direct Neural Architecture Search on Target Task and Hardware'. In: *International Conference on Learning Representations*. URL: `https://openreview.net/forum?id=HylVB3AqYm` (cit. on p. 32).

Carneiro, Gustavo, Jacinto C Nascimento and António Freitas (2011). 'The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods'. In: *IEEE Transactions on Image Processing* 21.3, pp. 968–982 (cit. on pp. 79, 80).

Cerri, Ricardo, Rodrigo C Barros and André CPLF De Carvalho (2014). 'Hierarchical multi-label classification using local neural networks'. In: *Journal of Computer and System Sciences* 80.1, pp. 39–56 (cit. on p. 54).

Chakraborty, Bidisha et al. (2018). '2-D myocardial deformation imaging based on RF-based nonrigid image registration'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 65.6, pp. 1037–1047 (cit. on p. 109).

Chen, Chen et al. (2020). 'Deep learning for cardiac image segmentation: A review'. In: *Frontiers in Cardiovascular Medicine* 7, p. 25 (cit. on p. 5).

Chen, Hao et al. (2016). 'Iterative multi-domain regularized deep learning for anatomical structure detection and segmentation from ultrasound images'. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 487–495 (cit. on p. 80).

Chen, Liang-Chieh, Maxwell Collins et al. (2018). 'Searching for efficient multi-scale architectures for dense image prediction'. In: *Advances in neural information processing systems*, pp. 8699–8710 (cit. on pp. 40, 48).

Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos et al. (2017). 'Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs'. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4, pp. 834–848 (cit. on p. 38).

Chen, Liang-Chieh, George Papandreou, Florian Schroff et al. (2017). 'Rethinking atrous convolution for semantic image segmentation'. In: *CoRR* abs/1706.05587 (cit. on pp. 36, 93).

Cheng, Kevin et al. (2018). '3D echocardiography: benefits and steps to wider implementation'. In: *British Journal of Cardiology* (cit. on p. 136).

Chollet, François (2017). 'Xception: Deep learning with depthwise separable convolutions'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258 (cit. on p. 42).

Chrabaszcz, Patryk, Ilya Loshchilov and Frank Hutter (2017). 'A downsampled variant of imagenet as an alternative to the cifar datasets'. In: *CoRR* abs/1707.08819 (cit. on p. 46).

Çiçek, Özgün et al. (2016). '3D U-Net: learning dense volumetric segmentation from sparse annotation'. In: *International conference on medical image computing and computer-assisted intervention*. Springer, pp. 424–432 (cit. on pp. 38, 49).

Cleve, Jayne and Marti L McCulloch (2018). 'Conducting a Cardiac Ultrasound Examination'. In: *Echocardiography*. Springer, pp. 33–42 (cit. on p. 16).

Coates, Adam et al. (2013). 'Deep learning with COTS HPC systems'. In: *International conference on machine learning*, pp. 1337–1345 (cit. on p. 51).

Colson, Benoıt, Patrice Marcotte and Gilles Savard (2007). 'An overview of bilevel optimization'. In: *Annals of operations research* 153.1, pp. 235–256 (cit. on p. 60).

Curiale, Ariel H, Gonzalo Vegas-Sánchez-Ferrero and Santiago Aja-Fernández (2016). 'Influence of ultrasound speckle tracking strategies for motion and strain estimation'. In: *Medical image analysis* 32, pp. 184–200 (cit. on p. 22).

D'hooge, Jan, Bart Bijnens et al. (2002). 'Echocardiographic strain and strain-rate imaging: a new tool to study regional myocardial function'. In: *IEEE transactions on medical imaging* 21.9, pp. 1022–1030 (cit. on p. 123).

D'hooge, Jan, Andreas Heimdal et al. (2000). 'Regional strain and strain rate measurements by cardiac ultrasound: principles, implementation and limitations'. In: *European Journal of Echocardiography* 1.3, pp. 154–170 (cit. on p. 123).

D'hooge, Jan, Elisa Konofagou et al. (2002). 'Two-dimensional ultrasonic strain rate measurement of the human heart in vivo'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 49.2, pp. 281–286 (cit. on p. 123).

D'hooge, Jan, J Schlegel et al. (2001). 'Evaluation of transmural myocardial deformation and reflectivity characteristics'. In: *2001 IEEE Ultrasonics Symposium. Proceedings. An International Symposium (Cat. No. 01CH37263)*. Vol. 2. IEEE, pp. 1185–1188 (cit. on p. 110).

Dandel, Michael and Roland Hetzer (2009). 'Echocardiographic strain and strain rate imaging—clinical applications'. In: *International journal of cardiology* 132.1, pp. 11–24 (cit. on p. 122).

Dandel, Michael, Hans Lehmkuhl et al. (2009). 'Strain and strain rate imaging by echocardiography-basic concepts and clinical applicability'. In: *Current cardiology reviews* 5.2, pp. 133–148 (cit. on pp. 23, 125).

Danu, Manuela, Costin Florian Ciuşdel and Lucian Mihai Itu (2020). 'Deep learning models based on automatic labeling with application in echocardiography'. In: *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, pp. 373–378 (cit. on p. 137).

De Luca, Valeria, Gábor Székely and Christine Tanner (2015). 'Estimation of large-scale organ motion in B-mode ultrasound image sequences: A survey'. In: *Ultrasound in medicine & biology* 41.12, pp. 3044–3062 (cit. on p. 109).

Dean, Jeffrey (2016). 'Large-scale deep learning for building intelligent computer systems'. In: (cit. on p. 28).

Deo, Rahul C et al. (2017). 'An end-to-end computer vision pipeline for automated cardiac function assessment by echocardiography'. In: *CoRR* (cit. on pp. 5, 52, 69).

Doi, Kunio (2007). 'Computer-aided diagnosis in medical imaging: historical review, current status and future potential'. In: *Computerized medical imaging and graphics* 31.4-5, pp. 198–211 (cit. on p. 51).

Domhan, Tobias, Jost Tobias Springenberg and Frank Hutter (2015). 'Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves'. In: *Twenty-Fourth International Joint Conference on Artificial Intelligence* (cit. on p. 46).

Dong, Suyu, Gongning Luo, Guanxiong Sun et al. (2016). 'A left ventricular segmentation method on 3D echocardiography using deep learning and snake'. In: *2016 Computing in Cardiology Conference (CinC)*. IEEE, pp. 473–476 (cit. on p. 6).

Dong, Suyu, Gongning Luo, Kuanquan Wang et al. (2018). 'A combined fully convolutional networks and deformable model for automatic left ventricle segmentation based on 3D echocardiography'. In: *BioMed research international* 2018 (cit. on p. 81).

Dufaux, Frederic and Fabrice Moscheni (1995). 'Motion estimation techniques for digital TV: A review and a new contribution'. In: *Proceedings of the IEEE* 83.6, pp. 858–876 (cit. on p. 111).

Ebadollahi, Shahram, Shih-Fu Chang and Henry Wu (2004). 'Automatic view recognition in echocardiogram videos using parts-based representation'. In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* Vol. 2. IEEE, pp. II–II (cit. on p. 52).

Elsken, Thomas, Jan Hendrik Metzen and Frank Hutter (2019). 'Efficient Multi-Objective Neural Architecture Search via Lamarckian Evolution'. In: *International Conference on Learning Representations*. URL: `https://openreview.net/forum?id=ByME42AqK7` (cit. on pp. 42, 46).

Elsken, Thomas, Jan Hendrik Metzen, Frank Hutter et al. (2019). 'Neural architecture search: A survey.' In: *J. Mach. Learn. Res.* 20.55, pp. 1–21 (cit. on pp. 41, 43–45, 59).

Elsken, Thomas, Jan-Hendrik Metzen and Frank Hutter (2018). 'Simple and efficient architecture search for convolutional neural networks'. In: URL: `https://openreview.net/forum?id=SySaJ0xCZ` (cit. on p. 46).

Ferraiuoli, Paolo et al. (2019). 'Measurement of in vitro cardiac deformation by means of 3D digital image correlation and ultrasound 2D speckle-tracking echocardiography'. In: *Medical Engineering & Physics* 74, pp. 146–152 (cit. on pp. 8, 122).

Feurer, Matthias and Frank Hutter (2019). 'Hyperparameter optimization'. In: *Automated Machine Learning*. Springer, Cham, pp. 3–33 (cit. on p. 41).

Fukuta, Hidekatsu and William C Little (2008). 'The cardiac cycle and the physiologic basis of left ventricular contraction, ejection, relaxation, and filling'. In: *Heart failure clinics* 4.1, pp. 1–11 (cit. on p. 4).

Gandhi, Sumeet et al. (2018). 'Automation, machine learning, and artificial intelligence in echocardiography: a brave new world'. In: *Echocardiography* 35.9, pp. 1402–1418 (cit. on p. 109).

Gao, Hang et al. (2009). 'A fast convolution-based methodology to simulate 2-dd/3-d cardiac ultrasound images'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 56.2, pp. 404–409 (cit. on p. 112).

Gao, Xiaohong et al. (2017). 'A fused deep learning architecture for viewpoint classification of echocardiography'. In: *Information Fusion* 36, pp. 103–113 (cit. on p. 52).

Garbin, Christian, Xingquan Zhu and Oge Marques (2020). 'Dropout vs. batch normalization: an empirical study of their impact to deep learning'. In: *Multimedia Tools and Applications*, pp. 1–39 (cit. on p. 31).

Garcia, Damien, Pierre Lantelme and Eric Saloux (2018). 'Introduction to speckle tracking in cardiac ultrasound imaging'. In: *Handbook of Speckle Filtering and Tracking in Cardiovascular Ultrasound Imaging and Video. Institution of Engineering and Technology*, pp. 571–598 (cit. on p. 7).

Gergonne, JD (1974). 'The application of the method of least squares to the interpolation of sequences'. In: *Historia Mathematica* 1.4, pp. 439–447 (cit. on p. 114).

Geyer, Holly et al. (2010). 'Assessment of myocardial mechanics using speckle tracking echocardiography: fundamentals and clinical applications'. In: *Journal of the American Society of Echocardiography* 23.4, pp. 351–369 (cit. on p. 3).

Ghelich Oghli, Mostafa et al. (2017). 'Left ventricle segmentation using a combination of region growing and graph based method'. In: *Iranian Journal of Radiology* 14.2 (cit. on p. 79).

Golemati, Spyretta, Aimilia Gastounioti and Konstantina S Nikita (2016). 'Ultrasound-image-based cardiovascular tissue motion estimation'. In: *IEEE Reviews in Biomedical Engineering* 9, pp. 208–218 (cit. on p. 3).

Gordienko, Yu et al. (2018). 'Deep learning with lung segmentation and bone shadow exclusion techniques for chest x-ray analysis of lung cancer'. In: *International Conference on Computer Science, Engineering and Education Applications*. Springer, pp. 638–647 (cit. on p. 38).

Greenspan, Hayit, Bram Van Ginneken and Ronald M Summers (2016). 'Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique'. In: *IEEE Transactions on Medical Imaging* 35.5, pp. 1153–1159 (cit. on p. 79).

Hall, John E (2016). 'Guyton and hall textbook of medical physiology (13e)'. In: *Elsevier* (cit. on p. 4).

He, Kaiming et al. (2016). 'Deep residual learning for image recognition'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778 (cit. on pp. 33–35, 39, 42).

Hesamian, Mohammad Hesam et al. (2019). 'Deep learning techniques for medical image segmentation: Achievements and challenges'. In: *Journal of digital imaging* 32.4, pp. 582–596 (cit. on p. 36).

Heyde, Brecht et al. (2012). 'Elastic image registration versus speckle tracking for 2-d myocardial motion estimation: A direct comparison in vivo'. In: *IEEE transactions on medical imaging* 32.2, pp. 449–459 (cit. on p. 109).

Hinton, Geoffrey E (2010). 'Rectified linear units improve restricted boltzmann machines vinod nair'. In: (cit. on p. 29).

Hinton, Geoffrey E and Ruslan R Salakhutdinov (2006). 'Reducing the dimensionality of data with neural networks'. In: *science* 313.5786, pp. 504–507 (cit. on p. 28).

Hinton, Geoffrey E, Nitish Srivastava et al. (2012). 'Improving neural networks by preventing co-adaptation of feature detectors'. In: *CoRR* abs/1207.0580 (cit. on p. 31).

Hinton, Robert B and Katherine E Yutzey (2011). 'Heart valve structure and function in development and disease'. In: *Annual review of physiology* 73, pp. 29–46 (cit. on p. 15).

Horton, Kenneth D (2010). 'Basic Ultrasound Views'. In: *Case Based Echocardiography*. Springer, pp. 13–21 (cit. on p. 18).

Howard, James P et al. (2020). 'Improving ultrasound video classification: an evaluation of novel deep learning methods in echocardiography'. In: *Journal of medical artificial intelligence* 3 (cit. on p. 69).

Huang, Gao et al. (2017). 'Densely connected convolutional networks'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708 (cit. on pp. 33, 35, 36, 42, 53).

Hui, Liu and Ye Xinhua (2020). 'A Novel Improved Multi-Point Matching Based Coronary Disease Quantitative Analysis for Speckle Tracking in Ultrasound Image'. In: *Journal of Medical Imaging and Health Informatics* 10.2, pp. 489–495 (cit. on pp. 8, 123).

Hutter, Frank, Lars Kotthoff and Joaquin Vanschoren (2019). *Automated machine learning: methods, systems, challenges*. Springer Nature (cit. on pp. 32, 40).

Ingul, Charlotte Bjork et al. (2005). 'Automated analysis of strain rate and strain: feasibility and clinical implications'. In: *Journal of the American Society of Echocardiography* 18.5, pp. 411–418 (cit. on p. 124).

Ioffe, Sergey and Christian Szegedy (2015). 'Batch normalization: Accelerating deep network training by reducing internal covariate shift'. In: *International conference on machine learning*. PMLR, pp. 448–456 (cit. on p. 30).

Jafari, Mohammad H et al. (2018). 'A Unified Framework Integrating Recurrent Fully-Convolutional Networks and Optical Flow for Segmentation of the Left Ventricle in Echocardiography Data'. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer, pp. 29–37 (cit. on p. 79).

Jaglan, Poonam, Rajeshwar Dass and Manoj Duhan (2019). 'A comparative analysis of various image segmentation techniques'. In: *Proceedings of 2nd International Conference on Communication, Computing and Networking*. Springer, pp. 359–374 (cit. on p. 78).

Jasaityte, Ruta, Brecht Heyde and Jan D'hooge (2013). 'Current state of three-dimensional myocardial strain estimation using echocardiography'. In: *Journal of the American Society of Echocardiography* 26.1, pp. 15–28 (cit. on p. 109).

Jeganathan, Jelliffe et al. (2017). 'Artificial intelligence in mitral valve analysis'. In: *Annals of cardiac anaesthesia* 20.2, p. 129 (cit. on p. 51).

Jégou, Simon et al. (2017). 'The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 11–19 (cit. on pp. 37, 79).

Jensen, Jørgen Arendt (1996). 'Field: A program for simulating ultrasound systems'. In: *10TH NORDICBALTIC CONFERENCE ON BIOMEDICAL IMAGING, VOL. 4, SUPPLEMENT 1, PART 1: 351–353*. Citeseer (cit. on p. 108).

Joos, Philippe et al. (2018). 'High-frame-rate speckle-tracking echocardiography'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 65.5, pp. 720–728 (cit. on pp. 8, 110, 123, 125).

Jordan, Michael I and Tom M Mitchell (2015). 'Machine learning: Trends, perspectives, and prospects'. In: *Science* 349.6245, pp. 255–260 (cit. on p. 27).

Kadappu, Krishna K and Liza Thomas (2015). 'Tissue Doppler imaging in echocardiography: value and limitations'. In: *Heart, Lung and Circulation* 24.3, pp. 224–233 (cit. on p. 21).

Kalam, Kashif, Petr Otahal and Thomas H Marwick (2014). 'Prognostic implications of global LV dysfunction: a systematic review and meta-analysis of global longitudinal strain and ejection fraction'. In: *Heart* 100.21, pp. 1673–1680 (cit. on pp. 8, 122).

Kalchbrenner, N, E Grefenstette and Philip Blunsom (2014). 'A convolutional neural network for modelling sentences'. In: *52nd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics (cit. on p. 28).

Kaur, Dilpreet and Yadwinder Kaur (2014). 'Various image segmentation techniques: a review'. In: *International Journal of Computer Science and Mobile Computing* 3.5, pp. 809–814 (cit. on p. 78).

Khamis, Hanan, Sara Shimoni et al. (2016). 'Optimization-Based Speckle Tracking Algorithm for Left Ventricle Strain Estimation: A Feasibility Study'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 63.8, pp. 1093–1106 (cit. on pp. 109, 110).

Khamis, Hanan, Grigoriy Zurakhov et al. (2017). 'Automatic apical view classification of echocardiograms using a discriminative learning dictionary'. In: *Medical image analysis* 36, pp. 15–21 (cit. on pp. 4, 51).

Kim, Jong-Nam et al. (2002). 'Fast full search motion estimation algorithm using early detection of impossible candidate vectors'. In: *IEEE Transactions on Signal Processing* 50.9, pp. 2355–2365 (cit. on p. 109).

Kim, Sungwoong et al. (2019). 'Scalable neural architecture search for 3d medical image segmentation'. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 220–228 (cit. on p. 49).

Kingma, Diederik P. and Jimmy Ba (2015). 'Adam: A Method for Stochastic Optimization'. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (cit. on pp. 61, 92, 94).

Klabunde, Richard (2011). *Cardiovascular physiology concepts*. Lippincott Williams & Wilkins (cit. on p. 16).

Klein, Aaron et al. (2017). 'Fast bayesian optimization of machine learning hyperparameters on large datasets'. In: *Artificial Intelligence and Statistics*. PMLR, pp. 528–536 (cit. on p. 46).

Konofagou, Elisa et al. (2011). 'Physiologic cardiovascular strain and intrinsic wave imaging'. In: *Annual Review of Biomedical Engineering* 13, pp. 477–505 (cit. on p. 1).

Krizhevsky, Alex, Ilya Sutskever and Geoffrey E Hinton (2012). 'Imagenet classification with deep convolutional neural networks'. In: *Advances in neural information processing systems*, pp. 1097–1105 (cit. on pp. 32–34).

Kumar, Ritwik et al. (2009). 'Echocardiogram view classification using edge filtered scale-invariant motion features'. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 723–730 (cit. on p. 52).

– (2010). 'Cardiac disease detection from echocardiogram using edge filtered scale-invariant motion features'. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, pp. 162–169 (cit. on p. 52).

Lang, Roberto M, Luigi P Badano et al. (2015). 'Recommendations for cardiac chamber quantification by echocardiography in adults: an update from the American Society of Echocardiography and the European Association of Cardiovascular Imaging'. In: *European Heart Journal-Cardiovascular Imaging* 16.3, pp. 233–271 (cit. on pp. 3, 50, 131).

Lang, Roberto M, Victor Mor-Avi et al. (2006). 'Three-dimensional echocardiography: the benefits of the additional dimension'. In: *Journal of the American College of Cardiology* 48.10, pp. 2053–2069 (cit. on p. 19).

Leclerc, Sarah, Erik Smistad, Andreas Østvik et al. (2020). 'LU-Net: A Multistage Attention Network to Improve the Robustness of Segmentation of Left Ventricular Structures in 2-D Echocardiography'. In: *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* 67.12, pp. 2519–2530 (cit. on p. 81).

Leclerc, Sarah, Erik Smistad, Joao Pedrosa et al. (2019). 'Deep learning for segmentation using an open large-scale dataset in 2D echocardiography'. In: *IEEE transactions on medical imaging* 38.9, pp. 2198–2210 (cit. on pp. 10, 79, 81, 85).

LeCun, Yann, Yoshua Bengio and Geoffrey Hinton (2015). 'Deep learning'. In: *nature* 521.7553, pp. 436–444 (cit. on p. 27).

LeCun, Yann, Léon Bottou et al. (1998). 'Gradient-based learning applied to document recognition'. In: *Proceedings of the IEEE* 86.11, pp. 2278–2324 (cit. on p. 33).

Leung, Dominic Y and Arnold CT Ng (2010). 'Emerging clinical role of strain imaging in echocardiography'. In: *Heart, lung and circulation* 19.3, pp. 161–174 (cit. on pp. 122, 123).

Leung, KY Esther et al. (2010). 'Probabilistic framework for tracking in artifact-prone 3D echocardiograms'. In: *Medical image analysis* 14.6, pp. 750–758 (cit. on p. 79).

Li, Ming, Shizhou Dong et al. (2020). 'Unified model for interpreting multi-view echocardiographic sequences without temporal information'. In: *Applied Soft Computing* 88, p. 106049 (cit. on p. 82).

Li, Ming, Chengjia Wang et al. (2020). 'MV-RAN: Multiview recurrent aggregation network for echocardiographic sequences segmentation and full cardiac cycle analysis'. In: *Computers in Biology and Medicine*, p. 103728 (cit. on p. 82).

Li, Ming, Weiwei Zhang et al. (2019). 'Recurrent Aggregation Learning for Multi-view Echocardiographic Sequences Segmentation'. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 678–686 (cit. on p. 138).

Liberato, Christiane Bezerra Rocha et al. (2020). 'Early left ventricular systolic dysfunction detected by two-dimensional speckle-tracking echocardiography in young patients with congenital generalized lipodystrophy'. In: *Diabetes, Metabolic Syndrome and Obesity: Targets and Therapy* 13, p. 107 (cit. on p. 6).

Lin, Ning, Weichuan Yu and James S Duncan (2003). 'Combinative multi-scale level set framework for echocardiographic image segmentation'. In: *Medical Image Analysis* 7.4, pp. 529–537 (cit. on p. 79).

Litjens, Geert et al. (2017). 'A survey on deep learning in medical image analysis'. In: *Medical image analysis* 42, pp. 60–88 (cit. on pp. 28, 32).

Liu, Chenxi, Liang-Chieh Chen et al. (2019). 'Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 82–92 (cit. on p. 79).

Liu, Chenxi, Barret Zoph et al. (2018). 'Progressive neural architecture search'. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 19–34 (cit. on pp. 42, 47, 79).

Liu, Hanxiao, Karen Simonyan and Yiming Yang (2019). 'DARTS: Differentiable Architecture Search'. In: *International Conference on Learning Representations*. URL: `https://openreview.net/forum?id=S1eYHoC5FX` (cit. on pp. 8, 32, 42, 47, 48, 89, 91).

Liu, Wenqi and Kun Zeng (2018). 'SparseNet: A sparse DenseNet for image classification'. In: *CoRR* abs/1804.05340 (cit. on p. 35).

Liu, Zhi and Jianwen Luo (2017). 'Performance comparison of optical flow and block matching methods in shearing and rotating models'. In: *Medical Imaging 2017: Ultrasonic Imaging and Tomography*. Vol. 10139. International Society for Optics and Photonics, p. 1013917 (cit. on pp. 7, 109).

Loizou, Christos P, Constantinos S Pattichis and Jan D'hooge (2018). *Handbook of Speckle Filtering and Tracking in Cardiovascular Ultrasound Imaging and Video*. Institution of Engineering and Technology (cit. on p. 20).

Long, Jonathan, Evan Shelhamer and Trevor Darrell (2015). 'Fully convolutional networks for semantic segmentation'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440 (cit. on p. 36).

Lu, Le et al. (2017). 'Deep learning and convolutional neural networks for medical image computing'. In: *Advances in Computer Vision and Pattern Recognition* (cit. on p. 80).

Lynch, Patrick J and CC Jaffe (2006). 'Medical Illustrations'. In: *Center for Advanced Instructional Media, Yale University School of Medicine: New Haven, CT, USA* (cit. on p. 4).

Mada, Razvan O et al. (2015). 'How to define end-diastole and end-systole?: impact of timing on strain measurements'. In: *JACC: Cardiovascular Imaging* 8.2, pp. 148–157 (cit. on p. 5).

Madani, Ali, Ramy Arnaout et al. (2018). 'Fast and accurate view classification of echocardiograms using deep learning'. In: *NPJ digital medicine* 1.1, pp. 1–8 (cit. on pp. 51, 53, 138).

Madani, Ali, Jia Rui Ong et al. (2018). 'Deep echocardiography: data-efficient supervised and semi-supervised deep learning towards automated diagnosis of cardiac disease'. In: *NPJ digital medicine* 1.1, pp. 1–11 (cit. on pp. 53, 136).

Marwick, Thomas H (2006). 'Measurement of strain and strain rate by echocardiography: ready for prime time?' In: *Journal of the American College of cardiology* 47.7, pp. 1313–1327 (cit. on p. 24).

Miller, Geoffrey F, Peter M Todd and Shailesh U Hegde (1989). 'Designing Neural Networks using Genetic Algorithms.' In: *ICGA*. Vol. 89, pp. 379–384 (cit. on p. 44).

Milletari, Fausto, Nassir Navab and Seyed-Ahmad Ahmadi (2016). 'V-net: Fully convolutional neural networks for volumetric medical image segmentation'. In: *2016 fourth international conference on 3D vision (3DV)*. IEEE, pp. 565–571 (cit. on p. 49).

Mitchell, Steven C et al. (2002). '3-D active appearance models: segmentation of cardiac MR and ultrasound images'. In: *IEEE transactions on medical imaging* 21.9, pp. 1167–1178 (cit. on p. 79).

Modin, Daniel, Ditte Madsen Andersen and Tor Biering-Sørensen (2018). 'Echo and heart failure: when do people need an echo, and when do they need natriuretic peptides?' In: *Echo research and practice* 5.2, R65–R79 (cit. on p. 17).

Mondillo, Sergio et al. (2011). 'Speckle-tracking echocardiography: a new technique for assessing myocardial function'. In: *Journal of Ultrasound in Medicine* 30.1, pp. 71–83 (cit. on pp. 3, 6).

Mor-Avi, Victor et al. (2011). 'Current and evolving echocardiographic techniques for the quantitative evaluation of cardiac mechanics: ASE/EAE consensus statement on methodology and indications endorsed by the Japanese Society of Echocardiography'. In: *European Journal of Echocardiography* 12.3, pp. 167–205 (cit. on pp. 23, 118).

Moradi, Shakiba et al. (2019). 'MFP-Unet: A novel deep learning based approach for left ventricle segmentation in echocardiography'. In: *Physica Medica* 67, pp. 58–69 (cit. on p. 81).

Myronenko, Andriy, Xubo Song and David J Sahn (2007). 'LV motion tracking from 3D echocardiography using textural and structural information'. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 428–435 (cit. on p. 14).

Narula, Sukrit et al. (2016). 'Machine-learning algorithms to automate morphological and functional assessments in 2D echocardiography'. In: *Journal of the American College of Cardiology* 68.21, pp. 2287–2295 (cit. on p. 51).

Negishi, Kazuaki et al. (2013). 'What is the primary source of discordance in strain measurement between vendors: imaging or analysis?' In: *Ultrasound in medicine & biology* 39.4, pp. 714–720 (cit. on p. 131).

Nikita, Konstantina S (2013). 'Atherosclerosis: the evolving role of vascular image analysis.' In: *Computerized medical imaging and graphics: the official journal of the Computerized Medical Imaging Society* 37.1, p. 1 (cit. on p. 1).

Noble, J Alison and Djamal Boukerroui (2006). 'Ultrasound image segmentation: a survey'. In: *IEEE Transactions on medical imaging* 25.8, pp. 987–1010 (cit. on p. 79).

O'Rourke, Maria C and Byron R Mendenhall (2019). 'Transesophageal Echocardiogram (TEE)'. In: *StatPearls [Internet]*. StatPearls Publishing (cit. on p. 19).

Oghli, Mostafa Ghelich, Alireza Fallahi et al. (2012). 'A novel method for left ventricle volume measurement on short axis MRI images based on deformable superellipses'. In: *International Joint Conference on Advances in Signal Processing and Information Technology*. Springer, pp. 101–106 (cit. on p. 79).

Oghli, Mostafa Ghelich, Ali Mohammadzadeh et al. (2018). 'A hybrid graph-based approach for right ventricle segmentation in cardiac MRI by long axis information transition'. In: *Physica Medica* 54, pp. 103–116 (cit. on p. 79).

Oktay, Ozan et al. (2017). 'Anatomically constrained neural networks (ACNNs): application to cardiac image enhancement and segmentation'. In: *IEEE transactions on medical imaging* 37.2, pp. 384–395 (cit. on p. 80).

Østvik, Andreas, Erik Smistad, Svein Arne Aase et al. (2019). 'Real-time standard view classification in transthoracic echocardiography using convolutional neural networks'. In: *Ultrasound in medicine & biology* 45.2, pp. 374–384 (cit. on pp. 51, 53, 136).

Østvik, Andreas, Erik Smistad, Torvald Espeland et al. (2018). 'Automatic myocardial strain imaging in echocardiography using deep learning'. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer, pp. 309–316 (cit. on p. 124).

Ostvik, Andreas et al. (2021). 'Myocardial function imaging in echocardiography using deep learning.' In: *IEEE Transactions on Medical Imaging* (cit. on p. 124).

Otey, M et al. (2006). 'Automatic view recognition for cardiac ultrasound images'. In: *International Workshop on Computer Vision for Intravascular and Intracardiac Imaging*, pp. 187–194 (cit. on p. 52).

Ouzir, Nora, Adrian Basarab, Olivier Lairez et al. (2018). 'Robust optical flow estimation in cardiac ultrasound images using a sparse representation'. In: *IEEE transactions on medical imaging* 38.3, pp. 741–752 (cit. on p. 111).

Ouzir, Nora, Adrian Basarab, Hervé Liebgott et al. (2017). 'Motion estimation in echocardiography using sparse representation and dictionary learning'. In: *IEEE Transactions on Image Processing* 27.1, pp. 64–77 (cit. on p. 124).

Park, Jin Hyeong et al. (2007). 'Automatic cardiac view classification of echocardiogram'. In: *2007 IEEE 11th International Conference on Computer Vision*. IEEE, pp. 1–8 (cit. on p. 51).

Paszke, Adam et al. (2017). 'Automatic differentiation in pytorch'. In: (cit. on pp. 64, 95).

Pavlopoulos, Harry and Petros Nihoyannopoulos (2008). 'Strain and strain rate deformation parameters: from tissue Doppler to 2D speckle tracking'. In: *The international journal of cardiovascular imaging* 24.5, pp. 479–491 (cit. on p. 24).

Petrank, Yael, Lingyun Huang and Matthew O'Donnell (2009). 'Reduced peak-hopping artifacts in ultrasonic strain estimation using the Viterbi algorithm'. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 56.7, pp. 1359–1367 (cit. on p. 110).

Pham, Hieu et al. (2018). 'Efficient neural architecture search via parameters sharing'. In: *International Conference on Machine Learning*. PMLR, pp. 4095–4104 (cit. on p. 47).

Porée, Jonathan et al. (2018). 'A dual tissue-doppler optical-flow method for speckle tracking echocardiography at high frame rate'. In: *IEEE transactions on medical imaging* 37.9, pp. 2022–2032 (cit. on p. 109).

Prada, Francesco et al. (2015). 'From grey scale B-mode to elastosonography: multimodal ultrasound imaging in meningioma surgery—pictorial essay and literature review'. In: *BioMed research international* 2015 (cit. on p. 21).

Prisant, L MICHAEL, LO Watkins and AA Carr (1984). 'Exercise stress testing.' In: *Southern medical journal* 77.12, pp. 1551–1556 (cit. on p. 19).

Raynaud, C et al. (2017). 'Handcrafted features vs ConvNets in 2D echocardiographic images'. In: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, pp. 1116–1119 (cit. on pp. 5, 78).

Real, Esteban et al. (2019a). 'Aging evolution for image classifier architecture search'. In: *AAAI Conference on Artificial Intelligence* (cit. on pp. 40, 42, 45, 46).

– (2019b). 'Regularized evolution for image classifier architecture search'. In: *Proceedings of the aaai conference on artificial intelligence*. Vol. 33, pp. 4780–4789 (cit. on pp. 32, 47, 79).

Reisner, Shimon A et al. (2004). 'Global longitudinal strain: a novel index of left ventricular systolic function'. In: *Journal of the American Society of Echocardiography* 17.6, pp. 630–633 (cit. on p. 22).

Rodriguez, Alvaro et al. (2014). 'Two-dimensional gel electrophoresis image registration using block-matching techniques and deformation models'. In: *Analytical biochemistry* 454, pp. 53–59 (cit. on pp. 8, 123).

Ronneberger, Olaf, Philipp Fischer and Thomas Brox (2015). 'U-net: Convolutional networks for biomedical image segmentation'. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 234–241 (cit. on pp. 28, 36, 37, 49, 53, 79, 80).

Sanfilippo, F et al. (2018). 'Left ventricular systolic function evaluated by strain echocardiography and relationship with mortality in patients with severe sepsis or septic shock: a systematic review and meta-analysis'. In: *Critical Care* 22.1, p. 183 (cit. on p. 125).

Santurkar, Shibani et al. (2018). 'How does batch normalization help optimization?' In: *Proceedings of the 32nd international conference on neural information processing systems*, pp. 2488–2498 (cit. on p. 31).

Shanazaman, Farhatafreen, Chandana Patel Harapriyasahoo and Ashimsauravsahoo Rajanjha (2015). 'IMPLEMENTING GMM-BASED HIDDEN MARKOV RANDOM FIELD FOR COLOUR IMAGE SEGMENTATION'. In: *International Journal Of Engineering Sciences & Research Technology*, pp. 208–212 (cit. on p. 78).

Shen, Dinggang, Guorong Wu and Heung-Il Suk (2017). 'Deep learning in medical image analysis'. In: *Annual review of biomedical engineering* 19, pp. 221–248 (cit. on pp. 27, 28).

Shrestha, Sirish and Partho P Sengupta (2018). 'Imaging heart failure with artificial intelligence'. In: (cit. on p. 27).

Siegersma, KR et al. (2019). 'Artificial intelligence in cardiovascular imaging: state of the art and implications for the imaging cardiologist'. In: *Netherlands Heart Journal*, pp. 1–11 (cit. on p. 51).

Simonyan, Karen and Andrew Zisserman (2015). 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (cit. on pp. 33, 34, 38).

Smiseth, Otto A et al. (2015). 'Myocardial strain imaging: how useful is it in clinical decision making?' In: *European heart journal* 37.15, pp. 1196–1207 (cit. on pp. 8, 24).

– (2016). 'Myocardial strain imaging: how useful is it in clinical decision making?' In: *European heart journal* 37.15, pp. 1196–1207 (cit. on p. 123).

Smistad, Erik, Andreas Østvik et al. (2017). '2D left ventricle segmentation using deep learning'. In: *2017 IEEE international ultrasonics symposium (IUS)*. IEEE, pp. 1–4 (cit. on p. 80).

Stoitsis, John et al. (2006). 'Computer aided diagnosis based on medical image processing and artificial intelligence methods'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 569.2, pp. 591–595 (cit. on p. 51).

Strubell, Emma, Ananya Ganesh and Andrew McCallum (2019). 'Energy and Policy Considerations for Deep Learning in NLP'. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3645–3650 (cit. on pp. 33, 52).

Sussillo, David and LF Abbott (2015). 'Random walk initialization for training very deep feedforward networks'. In: (cit. on p. 30).

SUZUKI, Kenji (2017). 'Machine learning in medical imaging before and after introduction of deep learning'. In: *Medical Imaging and Information Sciences* 34.2, pp. 14–24 (cit. on p. 27).

Szegedy, Christian, Wei Liu et al. (2015). 'Going deeper with convolutions'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9 (cit. on pp. 33, 34).

Szegedy, Christian, Vincent Vanhoucke et al. (2016). 'Rethinking the inception architecture for computer vision'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826 (cit. on p. 53).

Tabassian, Mahdi et al. (2018). 'Diagnosis of heart failure with preserved ejection fraction: machine learning of spatiotemporal variations in left ventricular deform-

ation'. In: *Journal of the American Society of Echocardiography* 31.12, pp. 1272–1284 (cit. on p. 27).

Tajbakhsh, Nima et al. (2016). 'Convolutional neural networks for medical image analysis: Full training or fine tuning?' In: *IEEE transactions on medical imaging* 35.5, pp. 1299–1312 (cit. on p. 80).

Tarroni, G, W Bai and M Sinclair (2017). 'Human-level CMR image analysis with deep fully convolutional networks'. In: *CoRR, vol. abs/1710.09289* (cit. on p. 51).

Tavakoli, Vahid et al. (2008). 'Adaptive multi-resolution myocardial motion analysis of b-mode echocardiography images using combined local/global optical flow'. In: *2008 2nd International Conference on Bioinformatics and Biomedical Engineering*. IEEE, pp. 2303–2306 (cit. on pp. 7, 109, 111).

Tenbrinck, Daniel et al. (2013). 'Histogram-based optical flow for motion estimation in ultrasound imaging'. In: *Journal of mathematical imaging and vision* 47.1-2, pp. 138–150 (cit. on p. 109).

Teske, Arco J et al. (2007). 'Echocardiographic quantification of myocardial function using tissue deformation imaging, a guide to image acquisition and analysis using tissue Doppler and speckle tracking'. In: *Cardiovascular ultrasound* 5.1, p. 27 (cit. on p. 122).

Thomas, James D and Luigi P Badano (2013). 'EACVI-ASE-industry initiative to standardize deformation imaging: a brief update from the co-chairs'. In: *European Heart Journal–Cardiovascular Imaging* 14.11, pp. 1039–1040 (cit. on pp. 7, 108, 112, 125).

Ting, KM (2010). *Confusion Matrix. Encyclopedia of Machine Learning* (cit. on p. 63).

Torkashvand, Paria, Hamid Behnam and Zahra Alizadeh Sani (2012). 'Modified optical flow technique for cardiac motions analysis in echocardiography images'. In: *Journal of medical signals and sensors* 2.3, p. 121 (cit. on p. 109).

Vanschoren, Joaquin (2019). 'Meta-learning'. In: *Automated Machine Learning*. Springer, Cham, pp. 35–61 (cit. on p. 41).

Vaseli, Hooman et al. (2019). 'Designing lightweight deep learning models for echocardiography view classification'. In: *Medical Imaging 2019: Image-Guided Procedures, Robotic Interventions, and Modeling*. Vol. 10951. International Society for Optics and Photonics, 109510F (cit. on p. 53).

Voigt, Jens-Uwe et al. (2014). 'Definitions for a common standard for 2D speckle tracking echocardiography: consensus document of the EACVI/ASE/Industry Task Force to standardize deformation imaging'. In: *European Heart Journal-Cardiovascular Imaging* 16.1, pp. 1–11 (cit. on pp. 7, 108, 109, 112, 125).

Voigt, Jens-Uwe et al. (2015). 'Definitions for a common standard for 2D speckle tracking echocardiography: consensus document of the EACVI/ASE/Industry Task Force to standardize deformation imaging'. In: *European Heart Journal-Cardiovascular Imaging* 16.1, pp. 1–11 (cit. on pp. 3, 108, 118).

Wang, Panqu et al. (2018). 'Understanding convolution for semantic segmentation'. In: *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, pp. 1451–1460 (cit. on p. 89).

Wei, Tao et al. (2016). 'Network morphism'. In: *International Conference on Machine Learning*, pp. 564–572 (cit. on p. 46).

Weng, Yu et al. (2019). 'Nas-unet: Neural architecture search for medical image segmentation'. In: *IEEE Access* 7, pp. 44247–44257 (cit. on p. 49).

Williams, Ronald J (1992). 'Simple statistical gradient-following algorithms for connectionist reinforcement learning'. In: *Machine learning* 8.3-4, pp. 229–256 (cit. on p. 44).

Wolf, Ivo et al. (2002). 'ROPES: A semiautomated segmentation method for accelerated analysis of three-dimensional echocardiographic data'. In: *IEEE transactions on medical imaging* 21.9, pp. 1091–1104 (cit. on p. 79).

World Health Organisation (2018). *The top 10 causes of death*. `https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death/`, Last accessed on 2020-01-06 (cit. on p. 1).

Wu, Hui et al. (2013). 'Echocardiogram view classification using low-level features'. In: *2013 IEEE 10th International Symposium on Biomedical Imaging*. IEEE, pp. 752–755 (cit. on p. 52).

Wu, Ke, Huazhong Shu and Jean-Louis Dillenseger (2014). 'Region and boundary feature estimation on ultrasound images using moment invariants'. In: *Computer methods and programs in biomedicine* 113.2, pp. 446–455 (cit. on p. 108).

Xie, Sirui et al. (2019). 'SNAS: stochastic neural architecture search'. In: *International Conference on Learning Representations*. URL: `https://openreview.net/forum?id=rylqooRqK7` (cit. on p. 47).

Yu, Fisher and Vladlen Koltun (2016). 'Multi-Scale Context Aggregation by Dilated Convolutions'. In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings* (cit. on p. 42).

Zela, Arber et al. (2018). 'Towards Automated Deep Learning: Efficient Joint Neural Architecture and Hyperparameter Search'. In: *ICML 2018 AutoML Workshop* (cit. on p. 46).

Zeng, Guodong et al. (2017). '3D U-net with multi-level deep supervision: fully automatic segmentation of proximal femur in 3D MR images'. In: *International workshop on machine learning in medical imaging*. Springer, pp. 274–282 (cit. on p. 38).

Zhang, Jeffrey et al. (2018). 'Fully automated echocardiogram interpretation in clinical practice: feasibility and diagnostic accuracy'. In: *Circulation* 138.16, pp. 1623–1635 (cit. on pp. 51, 52, 85).

Zhao, Hengshuang et al. (2017). 'Pyramid scene parsing network'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890 (cit. on p. 89).

Zhou, Shaohua Kevin et al. (2006). 'Image-based multiclass boosting and echocardiographic view classification'. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. IEEE, pp. 1559–1565 (cit. on p. 51).

Zhou, Zongwei et al. (2018). 'Unet++: A nested u-net architecture for medical image segmentation'. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer, pp. 3–11 (cit. on pp. 39, 40).

Zhu, Zhuotun et al. (2019). 'V-nas: Neural architecture search for volumetric medical image segmentation'. In: *2019 International Conference on 3D Vision (3DV)*. IEEE, pp. 240–248 (cit. on p. 49).

Zolgharni, Massoud et al. (2017). 'Automatic detection of end-diastolic and end-systolic frames in 2D echocardiography'. In: *Echocardiography* 34.7, pp. 956–967 (cit. on p. 109).

Zoph, Barret and Quoc V Le (2017). 'Neural architecture search with reinforcement learning'. In: URL: `https://openreview.net/forum?id=S1c2cvqee` (cit. on pp. 40, 42, 44, 47).

Zoph, Barret, Vijay Vasudevan et al. (2018). 'Learning transferable architectures for scalable image recognition'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8697–8710 (cit. on pp. 32, 40, 42, 44, 46, 79).

Zyuzin, Vasily et al. (2018). 'Identification of the left ventricle endocardial border on two-dimensional ultrasound images using the convolutional neural network Unet'. In: *2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*. IEEE, pp. 76–78 (cit. on p. 80).