

Deep Learning Using Convolutional Neural Networks in Clinical Cardiology

Dr James Howard

MA MB BChir MRCP

A thesis submitted to Imperial College London for the degree of Doctor of Philosophy

Statement of Originality

I hereby declare that the work presented in this thesis is my own, and that all else is appropriately referenced.

Copyright Declaration

‘The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-NonCommercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Acknowledgements

My journey in research started almost ten years ago when Professor (then Doctor) Darrel Francis (unfairly) accused me of lying on my CV and then offered me a job. It is his expert guidance and endless patience that has allowed me to develop into the rather peculiar mix of computer scientist and doctor I am today.

I also want to thank The Wellcome Trust and the Imperial 4i programme for funding these three years. I cannot overstate how much I've appreciated their flexibility. It has allowed me to explore my interests at the cutting edge of a field where things appear to be out of date within just a few years.

Dedication

I dedicate this thesis to all those who have supported me for these three years. My fiancée Alexia, who blessed me not only with her love, but also her eternal patience and understanding during those countless evenings and weekends when I needed to do “just a little bit more work”. My parents, Rob and Clare, who have never wavered once in their support in the last 34 years, even when I came bottom of Latin in 3rd year. And finally, my colleagues, who have made W12 a truly joyous place to be over the last three years.

Abstract

Computers have revolutionised modern medicine over the last two decades, with the advent of electronic health records and digital imaging. However, it has been difficult to develop computer programs which can assist doctors in interpreting these data. The traditional approach of programming a computer with an algorithm works well for simple decision making on low-dimensional data such as a series of blood tests. However, for highly complex data such as X-rays, echocardiography and magnetic resonance imaging scans, it has seemed impossible to develop adequately performing algorithms for even the simplest medical tasks.

Recently, however, a new paradigm in computer programming has emerged: machine learning. In machine learning, a computer learns how to perform a task without being explicitly programmed how to do it, and an increasingly popular approach to machine learning is ‘deep learning’ with neural networks. These neural networks somewhat resemble the animal brain in that they are made up of a series of neurons with adjustable connections between them. These connections are akin to synapses, and by tuning them optimally, computers can perform complex tasks such as image interpretation.

In this thesis I explore applications of neural networks to provide automated assistance for common cardiac image interpretation tasks:

- identifying the make and model of a cardiac device on a chest X-ray
- distinguishing successful from unsuccessful His bundle pacing, using an ECG
- identifying which view is represented on an echocardiogram video loop
- rapidly interpret the early anatomy sequences on a cardiac magnetic resonance (CMR) scan

For each of these tasks, I developed a specialised neural network design. They are all ‘convolutional neural networks’, which are inspired by the mammalian visual cortex, but they have different structures which I have selected to handle the different tasks.

First, I develop a convolutional neural network which can identify the exact model of pacemaker or defibrillator a patient has from their chest X-ray. The network is so effective that its performance exceeds that of cardiac electrophysiologists performing the same task. This system could speed up the diagnosis and treatment of patients with cardiac rhythm devices. After I published it and made it available online, doctors have reported using it successfully in their clinical practice.

Second, I develop a convolutional neural network which can discriminate between the ECG responses to His bundle pacing. It is able to identify when the procedure has been successful and is able to perform this task without requiring the intra-procedural electrograms and complex pacing manoeuvres that humans rely on to perform this task.

Third, I develop a series of neural networks which can classify echocardiogram videos according to what anatomical structures are depicted in them. This novel approach to ‘view classification’ more than halves the error rate of previous state-of-the-art methods.

Finally, I investigate whether deep learning of the anatomy sequences acquired in the first minutes of a cardiac magnetic resonance scan could provide early useful diagnostic information. I train a convolutional neural network to reconstruct a 3-dimensional model of the heart and major vessels from these early images and find it was able to accurately quantify cardiac chamber size, aortic diameter and identify the presence of pleural effusions. This system could be useful for identifying unexpected pathology, to allow optimisation of the scanning protocol within minutes of a scan commencing.

Table of Contents

Copyright Declaration.....	2
Acknowledgements.....	3
Dedication.....	4
Abstract	5
Table of Contents	7
List of Figures.....	16
List of Tables.....	28
1 Introduction	30
1.1 Cardiac implantable electronic device identification.....	32
1.1.1 Cardiac implantable electronic devices	32
1.1.2 CIED interrogation	32
1.1.3 Critical need for correct identification of CIED manufacturer	34
1.1.3.1 Current identification card systems are often inadequate.....	34
1.1.3.2 Chest X-ray device identification flow charts.....	37
1.1.4 My proposal for a neural network approach to CIED identification	40
1.2 Identification of successful His Bundle Pacing.....	42
1.2.1 The rationale for His bundle pacing	42
1.2.2 His bundle pacing has a high barrier to entry due to the need for skilled ECG interpretation.....	43
1.2.3 Convolutional neural networks are a possible solution.....	44
1.3 Echocardiogram view identification	45
1.3.1 Echocardiography and difficulties for deep learning.....	45

1.3.2	Echocardiography ‘views’	47
1.3.3	Current deep learning approaches for echocardiogram view identification.....	47
1.4	Automated analysis of cardiovascular magnetic resonance anatomy images	49
1.4.1	Rationale for an “early warning system”	49
1.4.2	The axial anatomy sequence	50
1.5	Aims of this thesis	52
2	Methods - Key machine learning & deep learning methodology	53
2.1	An introduction to machine learning	53
2.2	Simple forms of machine learning	54
2.3	Supervised and unsupervised learning.....	57
2.4	Neural networks	57
2.4.1	Advantages of neural networks over classical approaches such as logistic regression	62
2.4.2	Origin and utility of non-linearity.....	63
2.4.3	More layers allow inputs to be combined at different levels	66
2.4.4	Multi-layer perceptrons in cardiology, and their limitations.....	67
2.5	Convolutional Neural Networks.....	67
2.5.1	Inspiration from the mammalian brain.....	68
2.5.2	Convolutional neural networks learn by matching templates	69
2.5.3	Advantages of convolutional neural networks over classical neural networks	73
2.6	Convolutional neural networks beyond images	75
2.7	Convolutional neural networks’ strengths extend beyond classification: regression and segmentation.....	76
2.8	Training neural networks.....	78

2.9	Other strategies to improve convolutional neural network performance – reducing overfitting.....	84
2.10	Visualisation of learning - saliency mapping	90
2.11	The future of deep learning – optimism and scepticism	95
3	Methods – Software & Hardware	97
3.1	Software	97
3.1.1	Machine learning programming languages and libraries	97
3.1.2	Statistical programming.....	98
3.1.3	Integrated development environments.....	98
3.2	Deep learning hardware (graphical processing units)	101
4	Identification of cardiac implantable electronic devices from chest X-rays	104
4.1	Abstract.....	105
4.1.1	Background.....	105
4.1.2	Methods.....	105
4.1.3	Results.....	106
4.1.4	Conclusions.....	106
4.2	Introduction	107
4.2.1	Interrogation and reprogramming of cardiac implantable electronic devices.....	107
4.2.2	Current approaches for identification of CIEDs.....	107
4.3	Methods	109
4.3.1	Data extraction.....	109
4.3.2	Data processing.....	110
4.3.3	Convolutional neural network architecture and training	110
4.3.4	Visualisation of learning.....	111

4.3.5	Human expert performance using the flow chart	111
4.3.6	Statistical analysis	112
4.4	Results	113
4.4.1	Dataset.....	113
4.4.2	Stage 1: Neural network architecture development.....	117
4.4.3	Stage 2: Final neural network training and assessment	119
4.4.4	Comparison with human performance using the flow chart.....	123
4.4.5	Visualisation of learning using saliency mapping.....	125
4.5	Discussion	128
4.5.1	Neural network architecture greatly affects performance.....	128
4.5.2	Clinical applications	129
4.5.3	Saliency maps – human learning from machine learning.....	129
4.5.4	Study limitations.....	130
4.6	Conclusion.....	132
5	Identification of successful His Bundle Pacing using Convolutional Neural Networks	133
5.1	Abstract.....	134
5.1.1	Introduction.....	134
5.1.2	Methods.....	134
5.1.3	Results.....	134
5.1.4	Conclusions.....	135
5.2	Introduction	136
5.2.1	His bundle pacing is a novel form pacing	136
5.2.2	Successful His bundle pacing requires skilled ECG interpretation	137

5.2.3	Convolutional neural networks for classifying biological waveforms.....	139
5.3	Methods	141
5.3.1	Identification of cases.....	141
5.3.2	Data extraction and dataset creation.....	143
5.3.3	Classification of beats.....	144
5.3.4	Segmentation of beats	144
5.3.5	Neural network development	148
5.3.5.1	Re-implementation of the state-of-the-art convolutional neural network	148
5.3.5.2	The creation of novel 1-dimensional convolutional neural networks, adapted from established 2-dimensional convolutional neural networks	149
5.3.5.3	Re-implementation of the state-of-the-art RNNs.....	150
5.3.6	Neural network training.....	152
5.3.7	Visualisation of learning.....	152
5.3.8	Statistical analysis	153
5.4	Results	154
5.4.1	Study dataset.....	154
5.4.2	Performance of the different neural network architectures	157
5.4.3	Saliency mapping.....	160
5.5	Discussion	162
5.5.1	Clinical significance.....	162
5.5.2	CNNs outperform RNNs at classification of ECG responses to His bundle pacing	163

5.5.3	ResNet34 is superior to other network architectures, including larger ResNets and those derived from them	164
5.5.4	Saliency mapping not only highlights salient information, but may help identify errors	165
5.5.5	Study limitations.....	166
5.6	Conclusion.....	167
6	Echocardiogram view identification	168
6.1	Abstract.....	169
6.1.1	Background.....	169
6.1.2	Methods and Results	169
6.1.3	Conclusion	170
6.2	Introduction	171
6.3	Methods	175
6.3.1	Data sources	175
6.3.2	Data processing.....	175
6.3.3	Human image classification.....	176
6.3.3.1	Step 1 – Hierarchical manual labelling.....	178
6.3.3.2	Step 2 – Computer-assisted labelling.....	178
6.3.3.3	Reporting by a second physician - reproducibility of labels	185
6.3.4	Neural network details	185
6.3.4.1	Classical 2D CNN	185
6.3.4.2	Time-distributed 2D CNN.....	186
6.3.4.3	3D CNNs.....	187

6.3.4.4	Two-stream networks	187
6.3.5	Statistical analysis	189
6.4	Results	190
6.4.1	Dataset.....	190
6.4.2	The performance of the different neural network architectures.....	192
6.4.3	Inter-cardiologist agreement.....	195
6.5	Discussion	196
6.5.1	Temporal neural networks achieve half the error rate of classical 2D CNNs	196
6.5.2	Temporal neural networks show both a magnitude and pattern of errors similar to a human cardiologist	200
6.5.3	3D CNNs fail to surpass the performance of classical CNNs.....	200
6.5.4	Utility of accurate view identification	202
6.5.5	Study limitations.....	202
6.6	Conclusion.....	203
7	Automated analysis of cardiovascular magnetic resonance anatomy images	204
7.1	Abstract.....	205
7.1.1	Background.....	205
7.1.2	Methods.....	205
7.1.3	Results.....	205
7.1.4	Conclusions.....	206
7.2	Introduction	207
7.3	Methods	208
7.3.1	Data extraction.....	208

7.3.2	Data processing and labelling.....	210
7.3.3	Neural network design and training.....	212
7.3.4	3D heart model reconstruction.....	212
7.3.5	Statistical analysis	215
7.4	Results	216
7.4.1	Dataset.....	216
7.4.2	Segmentation performance.....	216
7.4.3	Diagnostic model performance	220
7.5	Discussion	229
7.5.1	Earlier & automated diagnosis – a comparison with previous approaches	229
7.5.2	Implications for reporting prioritization, supervision and patient safety	230
7.5.3	Study limitations.....	231
7.6	Conclusion.....	232
8	Synthesis.....	233
8.1	Convolutional neural networks provide options for automation of tasks which are difficult to solve with traditional algorithmic approaches.	234
8.2	Different problems are suited to different architectures of convolutional neural network	239
8.3	We now have many good neural networks: the bottleneck for future medical AI will be capturing appropriate expert labelling	241
8.3.1	Multiple stages of fewer choices are more efficient.....	243
8.3.2	Spotting odd ones out is even more efficient.....	243
8.4	Neural networks can point out diagnostically useful information	247
8.5	AI will play an increasing role in medicine, with far reaching consequences	248

8.5.1	Why AI will not replace doctors	249
9	Conclusions	251
10	Appendices	252
10.1	Publications arising from this PhD (2017-2020)	252
10.1.1	First author peer-reviewed publications	252
10.1.2	First author invited editorials	252
10.1.3	Second author peer-reviewed publications	253
10.1.4	Other co-author publications	254
10.2	Awards during this PhD	255
10.3	Software I have developed and released into the public domain during this PhD	256
10.3.1	BARD ECG segmentation & labelling software	256
10.3.2	CoPhy	256
10.3.3	PyMinim – A simple python minimisation algorithm	257
11	References	258

List of Figures

Figure 1-1 - An example identification card for a Medtronic CIED. Reproduced with permission from Medtronic Inc.	35
Figure 1-2 - Examples of pacemaker programmers from 4 different device manufacturers. Each of these is several kilograms in weight. Images reproduced with permissions from Boston Scientific, Abbott Medical, Biotronik Worldwide, Microport Corporation.	36
Figure 1-3 - Three CIEDs seen on chest X-ray. The left, centre and right images depict Medtronic Claria, Medtronic Advisa, and Biotronik Evia devices, respectively. The centre and right images are visually more similar, however, despite being of different manufacturers.	38
Figure 1-4 - The CaRDIA-X algorithm. This is the most recent manual algorithm for identifying CIEDs from chest X-rays. It comprises up to 7 steps, and the index publication reported accuracies of around 90%, though it is now 8 years out of date. Reproduced with permission from Heart Rhythm. 2011 Jun;8(6):915-22. ⁵	39
Figure 1-5 – LEFT: A slice from an axial anatomy planning sequence. In this slice the heart is visible, and it appears the right ventricle (A) and right atrium (B) are dilated. However, measurement of cardiac structures is not advised in these non-standard views, because the cardiac structures are not appropriately aligned with the scan planes, and the contrast between the left ventricular wall and blood pool is poor (C) compared with a dedicated 4 chamber sequence. In this picture, further diagnostic information is available, including imaging of the descending aorta (D) and evidence of a pleural effusion (E). RIGHT: An example image of a dedicated 4 chamber acquisition. The contrast between the left ventricular wall and blood pool is noticeably improved, and the four cardiac chambers are correctly imaged using an appropriate scanning plane, or “on axis”.....	51

Figure 2-1 - The relationship between age, cholesterol and mortality (probability of death)..... 56

Figure 2-2 - The simplest form of neural network. 5 input neurons, corresponding to 5 input variables feed into 1 output neuron, corresponding to the chance of death. The input neurons either increase (age, c-reactive protein, creatinine) or decrease (haemoglobin, ejection fraction) the activity of the output neuron, whose final value can be measured to provide a probability. The connections (weights) between the input neurons and the output neuron are adjusted to give the most accurate answer..... 59

Figure 2-3 - An example “classical” neural network, also termed a multi-layer perceptron (MLP). It is made up of a combination of stimulatory (green) and inhibitory (red) synapses, which process a series of inputs (age, haemoglobin level, c-reactive protein level, creatinine level and ejection fraction) to yield a probability of death. The final value, and therefore decision of the network, is decided by the relative strength of the synapses between the neurons (termed parameters, or ‘weights’). 61

Figure 2-4 - The rectified linear unit (ReLU) activation function. Values below 0 are set to 0; positive values are unchanged. 64

Figure 2-5 - A schematic of a simple neural network with two hidden units between an input and output neuron. The hidden units each comprise a neuron and a rectified linear unit (ReLU). These take a maximum of 0 and an input value, depicted as “ $\max(0, \text{input})$ ”. With just two neurons, we can create a small network which calculates the absolute value of an input value (yellow graph), or something which approximates a sigmoid function (blue graph)..... 65

Figure 2-6 - A schematic of a two-layer neural network designed to identify whether an image is of a nought or a cross. In the top row, we can see the image of the nought contains areas which match each of the four kernels in the convolutional layer. The cross image (bottom row), however, doesn’t contain any features that match the vertical or horizontal kernels. The fully

connected layer allows the strength of these detections to be translated into predictions; the weights connecting the vertical and horizontal kernel matches to the ‘nought’ prediction are stimulatory, but they are inhibitory for the ‘cross’ class..... 71

Figure 2-7 - A schematic of a neural network used to identify echocardiographic views. Early convolutional layers in this network identify basic features such as edges, whilst later layers combine these features to identify anatomical structures. The presence or absence of these structures leads to a final prediction by the network..... 72

Figure 2-8 - CNNs can be used for classification, regression and segmentation problems. Classification problems may involve deciding which class an image corresponds to (left panel). Regression tasks involve predicting one or more continuous values from an image, which could be used, for example, to locate a bounding box (middle panel). In segmentation tasks, the neural network may classify each pixel within an image as one of several classes (right panel; LV = left ventricle, RV = right ventricle)..... 77

Figure 2-9 - The steps involved in training a neural network. **Step 1:** We load 4 examples of training data and the correct answers, as a batch. **Step 2:** The batch is fed into the neural network, which makes predictions for each of the 4 examples. **Step 3:** We calculate the loss by comparing the neural network’s predictions with the labels, using the chosen loss function. **Step 4 & 5:** The optimizer uses the loss to adjust each of the weights in the neural network, with the size of the adjustment set by the user-supplied learning rate. **Step 6:** The cycle returns to Step 1, repeating until we have finished training..... 82

Figure 2-10 - A single picture of a pacemaker on a chest X-ray which has been randomly augmented. 16 different examples are shown, with random amounts of cropping, rotation, skewing and brightening. The black border reflects areas which lie outside of the original image. 86

Figure 2-11 - A schematic showing two otherwise identical neural networks, one of which uses dropout. During training, there is a certain percentage (e.g. 40%) that any single neuron in the hidden layers will be “dropped”, and not contribute towards the neural network’s outputs. The neurons dropped changes with every batch. This punishing the neural network for relying on a small population of neurons to make the final decision, as there is a 40% chance of these not being present during training time. During testing time, however, dropout is disabled, so all neurons can be used to come to a decision. Dropout can be applied to any layer of the neural network, though in practice one would not normally apply it to the input or output neurons as the former would reduce 40% of the input data, and the latter would prevent 40% of the categories being predictable by the network..... 89

Figure 2-12 - A saliency map of a Boston Ingenio pacemaker present on a chest X-ray. The left side of the image shows the original chest X-ray. The right side of the image shows the saliency map overlying the original image. Royal blue regions represent pixels which are contributing little towards the neural network’s decision, whilst brighter regions represent pixels which are greatly influencing the neural network’s decision that this is a Boston Ingenio device. 92

Figure 3-1 - An example of the PyCharm integrated development environment (IDE). The top left panel of the screen shows files that are part of the current project. The top right of the screen shows the Python code contained in a selected file. The bottom left shows the status of a program which is currently running. The bottom right shows the data currently being used by that program. 99

Figure 3-2 - The RStudio integrated development environment (IDE). The organisation is similar to that of PyCharm, with source code visible on the top left of the screen. The bottom left shows the ‘output’ of the program as it runs. The top right panel shows the current variables in the computer’s memory that contain data. The bottom right demonstrated a plot that I have created using the visible code, and which forms part of section 7.4.3..... 100

Figure 3-3 - Example of an Nvidia graphic processing unit (GPU) - the RTX Titan model is shown. Image Courtesy of Nvidia Corporation, California, United States.	102
Figure 3-4 - The workstation used by the Student. Two RTX Titan graphical processing units (GPUs) are visible below the large central processing unit (CPU) cooler. The power supply at the bottom of the case shows the current power draw for the system in Watts.	103
Figure 4-1 - Study design flowchart. The study was designed in three phases, comprising data collection, development of the neural network, and assessment of the network. The development of the neural network was broken down into two stages. Stage 1 involved selecting the optimal network design. Stage 2 involved training the ‘final’ model against the entire training set, and then assessing its performance on the unseen ‘test set’, allowing comparison versus humans...	116
Figure 4-2 - Neural network performance. Confusion matrix showing the accuracy of the network in predicting the correct manufacturer of devices. BIO=Biotronik; BOS=Boston Scientific; MDT=Medtronic; SOR=Sorin; STJ=St. Jude.	120
Figure 4-3 - Confusion matrix showing the accuracy of the network in predicting the correct model of devices. Class names ending in ‘...’ refer to those including more than 1 device type with identical appearance. BIO=Biotronik; BOS=Boston Scientific; MDT=Medtronic; SOR=Sorin; STJ=St. Jude.	121
Figure 4-4 - The single case where the neural network wrongly predicted the manufacturer of a device. The device in the picture of interest (left side) is that of a Medtronic Adapta. The neural network incorrectly predicted this device to be a Sorin Reply device.....	122
Figure 4-5 - Comparison between neural network and human performance. Bar plot showing comparative accuracy for identifying the manufacturer of devices across the five human	

reporters and the neural network. P values are for superiority of the neural network above the median and best human graders. 124

Figure 4-6 - Four images, depicting two Advisa devices, and two AT500 devices. Panel A is an Advisa. We invite readers to identify which other panel (B, C or D) is also an Advisa. The other two are AT500s. Additionally, how would you advise others to make the same distinction?..... 126

Figure 4-7 - Saliency plots from the neural network which could help guide us where to look. The answer to the question in Figure 4-6 is panel C. Saliency plots reveal that the network is focusing on a feature present in the AT500s (red circles) which is absent in the Advisa devices. Having this pointed out by the network makes it now easy to return to Figure 4-6 and correctly categorize them. These example images also demonstrate the neural network's ability to deal with dramatic differences in image quality, X-ray penetration and orientation..... 127

Figure 5-1 - The mechanisms and ECG findings of the three capture types seen during attempted His bundle pacing. Stim-V = interval between start of pacing spike and start of QRS; Stim-QRS_{end} = interval between start of pacing spike and end of QRS; QRSD = duration of QRS complex. Adapted with permission from Arnold & Howard et al.⁷⁸ 142

Figure 5-2 – The segmentation interface of the software I developed to allow manual segmentation of individual ECG beats. The buttons on the left hand side indicate whether a segment of the ECG has been segmented (green) or not (amber). Segmentation is done using blue slides which can be stretched over a region of interest of the ECG trace..... 146

Figure 5-3 - The data export interface of the software I developed to allow batch exporting of labelled ECGs. Certain protocols and start/stop boundaries can be chosen in the bottom left corner, to allow specific areas of specific ECGs to be extracted. The exported data will then comprise a database from which a neural network can be trained..... 147

Figure 5-4 - Two examples of ResNet networks. Panels A&B show ResNet34. This network comprises 34 layers (panel B), which is made up of a series of “basic blocks” (panel A), which are two sets of convolutional layers of size 3 (dark blue), a batch normalisation layer, and a rectified linear unit (ReLU). Importantly, there is a “skip” or “residual” connection across the two layers, which allows data to bypass the layers if this is advantageous during training. Larger ResNet networks, such as ResNet50 (panels C&D), use a different building block, termed a “bottleneck block” (panel C), which comprises three layers. The first and third layer are convolutional layers of size 1 (green) rather than 3. This allows more layers to be used, allowing the network to be deeper, without the number of weights within the networks being prohibitively large. 151

Figure 5-5 - Study flowchart. Individual patients were randomised between the training and testing datasets, with the ECGs extracted from each case. The training dataset was further divided into ‘folds’, with three neural networks trained for each architecture. *Fewer ECGs per patient were used for the testing dataset, to ensure a balanced number of unique ECG appearances across cases (see section 5.3.2, page 141). 155

Figure 5-6 - Confusion matrix for the best performing neural network. ResNet34 demonstrated 75% accuracy (kappa 0.588) in classifying beats across the three classes; MOC = myocardial-only capture; NS-HBP = non-selective His bundle pacing; S-HBP = selective His bundle pacing... 159

Figure 5-7 - Saliency maps demonstrating activity for chest leads. Dark blue areas are more salient. This is an indication of which parts of the ECG are contributing most towards the network's decision. 161

Figure 6-1 - The four different types of neural network architectures used in this study, along with the lowest error rate of each model within each group. The best-performing neural network was a “two-stream” network using both spatial and optical flow inputs, with a corresponding error rate of only 3.9%. Conv = Convolutional layer; Batch Norm = Batch Normalisation layer;

ReLU = Rectified linear unit layer. Reproduced with permission from Howard et al. (J Med. Artif Intell, 2019).⁹⁴..... 174

Figure 6-2 - The 14 echocardiographic views, with example echocardiographs and schematics below. A2CH = apical 2 chamber; A3CH = apical 3 chamber; A4CH = apical 4 chamber; A5CH = apical 5 chamber; Ao = Aorta; AV = aortic valve; IAS = interatrial septum; LA = left atrium; LV = left ventricle; PLAX = parasternal long axis; PS = parasternal; PV = pulmonary valve; RA = right atrium; RV = right ventricle; TV = tricuspid valve. Reproduced with permission from Howard et al. (J Med. Artif Intell, 2019).⁹⁴..... 177

Figure 6-3 - A simplified hierarchical model like that used in the rapid reporting software. Each level of hierarchy represents a stage of labelling for a video. This approach ensures only a maximum of 5 options are presented to the user for any video, facilitating rapid and accurate labelling. In practice, I used a total of 14 views, rather than the 6 shown on this diagram..... 180

Figure 6-4 - An example of the rapid labelling software I have created showing an example video playing with buttons for the appropriate categories. In this image, a view is being classified using the top level or hierarchy (where the probe is positioned), rather than the precise view. This minimises the number of choices available to the human labeller. Each category can be clicked as a button or selected by pressing the appropriate number (1 to 5 in the example above) on the reviewer's keyboard. The example video is a parasternal short axis video (PSAX) focused on the pulmonary valve, and so the user should either click the 'PSAX' button, or press '2' on their keyboard..... 181

Figure 6-5 – Another example of the rapid labelling software. Here, only parasternal short axis videos (labelled using the previous step) are shown to the human labeller, and videos are now classified according to the exact view depicted. This again limits the number of options from which the human must choose, and greatly simplifies the task. In this example, the video shows

the view is focused on the aortic valve (AV), and so the user should either click the ‘PSAX_AV’ button, or press ‘1’ on their keyboard..... 182

Figure 6-6 - Computer-assisted labelling. In this example, the computer has found 12 videos which it believed are Apical 4 Chamber videos (A4CH). The reporter reviews these, and notes a single mistake; the third video on the top row is actually an Apical 5 Chamber. The reporter clicks on this and marks the video for manual relabelling at a later date. The remaining 11 videos enter the dataset with the label ‘Apical 4 Chamber’. 183

Figure 6-7 - Confusion matrices for the best-performing classical CNN model (A), time-distributed model (B), 3D CNN (C) and two-stream network (D). The improvement associated with using the two-stream network versus the classical CNN is shown in panel E. The inter-human agreement confusion matrix is shown in panel F. 194

Figure 6-8 - A comparison of a ‘PLAX inflow’ view of the tricuspid valve (A) and a ‘parasternal [aortic and] pulmonary valves’ (B) view of the pulmonary valve (also termed ‘RV outflow’). These views are almost indistinguishable in a still image. However, when viewed as a video, the echocardiograph on the left clearly demonstrates the valve opening upwards (inwards; see arrows), allowing blood flow into the heart through the tricuspid valve, whereas the echocardiogram on the right shows the valve leaflets opening downwards (outwards; see arrows) allowing flow out of the heart. Misclassifications of these classes were common using classical 2D CNNs, but are almost eliminated by employing temporal models such as the ‘two stream’ networks. Saliency mapping can be used to visualise how the features from the pulmonary valve video contribute towards the two-stream network’s decision. In panels C and D, the important features leading to the classification are highlighted in cyan. Panel C shows the spatial arm of the network appears to use the anatomical borders of the major cardiac structures present (pulmonary artery and left ventricle). Panel D, however, shows the decision of the temporal arm of the network is overwhelmingly influenced by the optical flow data of the valve itself. 198

Figure 7-1 - Study design flow chart & data flow. The study comprised 575 studies across 2 manufacturers, which were split into a training and testing set. The training set was used to train an AI which could segment each slice of the anatomy sequence. Then, each scan in the testing set was analysed to allow creation of a 3D model of a patient's thorax. This 3D model was then analysed. 3D = 3-dimensional. DCM = dilated cardiomyopathy. HCM = hypertrophic cardiomyopathy. LV = left ventricle, RV = right ventricle. 209

Figure 7-2 - Reconstruction of a 3D heart is possible by segmenting each slice within the axial anatomy stack. The slices can be combined through trilinear interpolation into a 3D model of the cardiac chambers and great arteries. This model can be used to make anatomical measurements, such as end-diastolic volumes and vessel diameters..... 214

Figure 7-3 - A Human label and the AI's corresponding segmentation prediction. The figure shows a comparison between the human labels of a particular axial anatomical planning slice, and those predicted by the network for that slice..... 219

Figure 7-4 - The relationship between the neural network's chamber size predictions for body surface area-indexed left ventricular end-diastolic volume (LVEDVi) versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively. Patients with known dilated cardiomyopathy (DCM) are shown as crosses. 222

Figure 7-5 - The relationship between the neural network's chamber size predictions for body surface area-indexed right ventricular end-diastolic volume (RVEDVi) versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively..... 223

Figure 7-6 - The relationship between the neural network's predictions for body surface area-indexed left ventricular mass (LVMi) versus those extracted from the final report. Green zones

and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively. 224

Figure 7-7 - The relationship between the neural network's predictions for left ventricular (LV) mass to volume ratio versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively. 225

Figure 7-8 - The relationship between the neural network's body surface area-indexed ascending aortic diameter predictions versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively. 226

Figure 7-9 – The relationship between the predicted pleural fluid volume and the diagnosis of a pleural effusion. The blue curve represents the logistic regression model fitted. 227

Figure 7-10 - Receiver operating characteristic (ROC) curve for pleural effusion identification. The area under the curve (AUC) is 0.906..... 228

Figure 8-1 - Accuracy on the ImageNet image classification challenge by year. Data from <http://www.image-net.org/>. In 2012, the AlexNet convolutional neural network (CNN) posted a dramatic reduction in the error rate above all previous approaches, and CNNs have consistently improved upon this performance since. In 2015, CNNs surpassed human performance (green dotted line). 235

Figure 8-2 - Is this a cat? Reproduced from <https://www.flickr.com/photos/27587002@N07/5170590074> under a CC BY 2.0 license. ... 237

Figure 8-3 – Labelling software I have recently developed for an ongoing research project, involving the segmentation of the endocardial and epicardial left ventricular surfaces on cardiac MRI images..... 245

Figure 8-4 - Our an easy-to-use web-based interface to annotate medical images, which is currently in development. The system is divided into a labelled area (blue square) and an information area, showing that user’s statistics, compared with those of other users (red square). Fiducial points on echocardiograms can be labelled either using either a mouse, or a touch screen interface such as that on a phone or tablet, by dragging the circles and their attached crosshairs. 246

Figure 10-1 - A screenshot of the CoPhy coronary physiology suite..... 257

List of Tables

Table 2-1 - How each of the training, validation and testing datasets influence the network, and how closely the network's performance on these datasets reflect its "real world" performance.	81
Table 4-1 - Distribution of classes across the entire dataset, broken down by device type, manufacturer and model. Visually identical model names (middle column) are merged into 'model groups' (right column).....	115
Table 4-2 - Results of Stage 1, in which the 5 architectures are compared having been trained on only three quarters of the training data at a time. Performance of five network designs. Loss is a measure of inaccuracy which gives penalizes confident wrong predictions more than unconfident ones.	118
Table 5-1 - Patient characteristics. Values are mean $\pm$ standard deviation (range) for age and ejection fraction. Otherwise n and percentage are given.	156
Table 5-2 - Results on the testing dataset for the different neural network architectures assessed.	158
Table 6-1 - Baseline characteristics of study-level and video-level parameters. Categorical data is shown as numbers (%); continuous data are shown as mean (standard deviation). * 273 and 93 videos from the training and testing sets, respectively, were unable to be classified, and do not contribute towards the class percentages; this is because they showed two or more views during the full video loop, or neither of the human operators were able to identify sufficient landmarks to classify the video.....	191
Table 6-2 - Table demonstrating the total number of trainable parameters, accuracy on the test set, and associated Cohen's kappa for each of the networks on the test set. * = This figures	

reflect the accuracy of the temporal network alone, which goes on to form one stream of each of the two-stream networks assessed. TD = Time-distributed.	193
Table 7-1 - Anatomical planning sequence parameters by manufacturer.....	211
Table 7-2 - Characteristics of the included studies in the training and testing sets. Values are mean (standard deviation) for continuous variables, and count (percentage) for categorical variables. LVEF = left ventricular ejection fraction. LVEDVi = Body-surface area-indexed left ventricular end-diastolic volumes. RVEDVi = Body-surface area-indexed right ventricular end-diastolic volumes. LVMi = Body-surface area-indexed right ventricular end-diastolic volumes. HCM = hypertrophic cardiomyopathy. DCM = dilated cardiomyopathy.....	217
Table 7-3 - Slices included in the training and testing sets. Values are n (%). The Dice coefficient reflects the segmentation accuracy of the neural network on the validation set.	218

1 Introduction

Modern medicine has entered the information age, with a wealth of data in the form of electronic health records, biometrics, multi-omics and biomedical imaging. However, only in the last few years have we developed techniques which can assist doctors in interpreting these complex data.¹

The traditional algorithmic approach to computer programming, where we provide a computer with an explicit series of steps to follow, has generally failed to provide useful results when applied to the field of image processing. In the last few years, however, machine learning has become an increasingly popular approach to these complex problems. In machine learning, we do not translate our human understanding of a task directly into code, but instead create computer programs with the ability to learn from many hundreds or thousands of examples. Through repeated exposure to the data along with the correct result, the computer continuously adjusts its settings so that it can better perform a task. The aim is that eventually the computer program can rival the performance of a human at a task, or even exceed it.

The most exciting approach to machine learning is using neural networks. These are computer programs which are made up of a series of computational ‘neurons’ with synapses between, akin to the animal brain. The strengths of these synapses are continuously adjusted, allowing the neural network to learn how to perform a task.

In this thesis I will explore several applications of neural networks in clinical cardiology.

In this chapter I will explore four separate clinical problems that I have identified that I believe neural networks can revolutionise.

I will then describe the key background and methodology of machine learning using neural networks in Chapter 2, and detail the specific methodology (software and hardware) I used in Chapter 3.

In chapters 4, 5, 6 and 7 I will describe my experiments across four specific clinical applications of neural networks.

In chapter 8 I will integrate my findings and draw my conclusions.

1.1 Cardiac implantable electronic device identification

This section serves as a preamble to Chapter 4 - Identification of cardiac implantable electronic devices from chest X-rays.

1.1.1 Cardiac implantable electronic devices

Cardiac implantable electronic devices (CIEDs) include pacemakers, cardiac resynchronisation therapy (CRT) devices, implantable cardioverter-defibrillators (ICDs) and implantable loop recorders (ILRs).² These devices are able to monitor (ILRs) and alter (pacemakers, CRTs & ICDs) the electrical functioning of the heart. They are used increasingly frequently in patients with suspected or known arrhythmia, and also in patients with heart failure with abnormal electrical functioning of the heart.

The numbers of CIEDs implanted each year continues to increase, with over 40 thousand devices inserted every year in the United Kingdom alone,³ and a million internationally.⁴ The increasing prevalence of patients with CIEDs means these devices are increasingly encountered in the emergency setting.

1.1.2 CIED interrogation

When a patient presents to hospital in an emergency with a CIED it is often important to “interrogate” the device. This process involves taking a proprietary manufacturer-specific piece of equipment to the patient’s bedside and holding an interrogation wand over the CIED.

Interrogation allows diagnostic information to be recovered about any recent arrhythmia and their current heart rhythm. For pacemakers, CRTs and ICDs, the therapies that they deliver during arrhythmias can also be adjusted during the interrogation process.

There are numerous clinical scenarios when interrogation may play a pivotal role in the emergency presentation of a patient with a CIED:

1. In a patient presenting with syncope, interrogation of a CIED will provide important information about whether this was arrhythmic in origin, and whether it was the consequence of a tachyarrhythmia, or a bradyarrhythmia, which are treated entirely differently.
2. In a patient presenting with a bradyarrhythmia despite a pacemaker device, there are numerous possible explanations for the device failing to pace that patient to a normal heart rate. One explanation is an increase in the pacing threshold needed to depolarise the myocardium, meaning the pacemaker is not delivering enough energy to pace the patient. Another explanation is electrical noise due to a problem with the pacing lead, meaning the pacemaker inappropriately detects a higher heart rate. Finally, the pacemaker's battery may have failed. Interrogation of the device in this setting will easily discriminate between the former two, and complete failure to interrogate indicates the third.
3. In a patient presenting with repeated defibrillation shocks from their ICD, these may either be appropriate (there is a true underlying arrhythmia which must be treated immediately) or inappropriate (the device is wrongly classifying a normal or less dangerous arrhythmia as one which requires defibrillation). Interrogation will allow the healthcare team to discriminate between these two explanations. Furthermore, in the setting of inappropriate shocks, interrogation also allows the ICD to be temporarily deactivated, so that further shocks are administered.

1.1.3 Critical need for correct identification of CIED manufacturer

For these reasons, rapid interrogation of CIEDs in patients presenting with symptoms of arrhythmia is often imperative in delivering optimal care. However, because each CIED requires manufacturer-specific interrogation hardware, this that historically relied on the patient being able to inform the clinical team of the type of device they have been implanted with. For this reason, patients are given identification cards at the time of their implant (Figure 1-1).

1.1.3.1 Current identification card systems are often inadequate

Despite patients being given identification cards at the time of their pacemaker implant, up to 80% of physicians have reported difficulties in identifying devices.⁵ This is unsurprising when one must consider patients presenting with arrhythmias or device malfunctions are frequently presenting with very acute symptoms, and so may not have their belongings with them.

Unfortunately, guessing the manufacturer is not ideal, because each manufacturer's CIEDs use a manufacturer-specific interrogation device. These are heavy and bulky (Figure 1-2) and typically stored in a hospital's cardiac outpatient pacing department, where they are in heavy use.

These issues mean that the cardiology team is frequently unable to bring the correct device programmer to the patient's bedside initially, unless they either guess, or arrange for several staff to bring several programmers to the patient's bedside.




First name Last name
Address line 1
Address line 2
City, State Zip code, Country (Non US)

Printed:
01-JAN-2017

Implant Date	Serial Number	Model Number
XX-XXX-XXXX	XXXXXXXXXX	XXXXXX
XX-XXX-XXXX	XXXXXXXXXX	XXXXXX
XX-XXX-XXXX	XXXXXXXXXX	XXXXXX
XX-XXX-XXXX	XXXXXXXXXX	XXXXXX
XX-XXX-XXXX	XXXXXXXXXX	XXXXXX

3330947
Medical Device Identification

For changes, contact Medtronic at www.medtronic.com or 1-800-551-5544 (for US only)

I have a/an XXXX implanted

My device may trigger metal detection systems

For Medical Questions or Emergency, Call:
First Name Last Name, MD (000) 000-0000

UC20151011A 0N © 2016 Medtronic

Figure 1-1 - An example identification card for a Medtronic CIED. Reproduced with permission from Medtronic Inc.



Figure 1-2 - Examples of pacemaker programmers from 4 different device manufacturers. Each of these is several kilograms in weight. Images reproduced with permissions from Boston Scientific, Abbott Medical, Biotronik Worldwide, Microport Corporation.

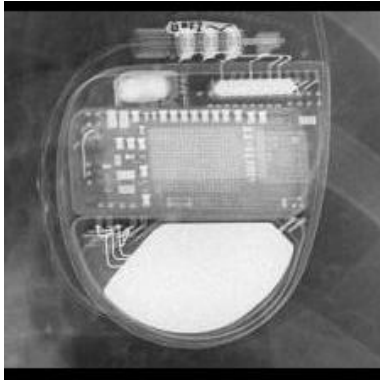
1.1.3.2 Chest X-ray device identification flow charts

In principle, the chest X-ray, that is often performed routinely in emergency department patients, contains the information needed to identify the CIED. A physician could identify the CIED by its appearance on the chest X-ray.

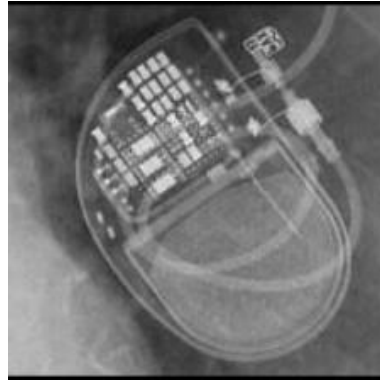
Unfortunately, the X-ray appearance of a single device manufacturer is not stereotyped; two different models from a single manufacturer can look markedly different on X-ray (Figure 1-3), making this process very difficult.

Due to these difficulties, flow chart approaches have been developed. The CaRDIA-X algorithm is one example of this, where staff can follow a stepwise examination of the CIED's X-ray appearance to allow identification of the relevant manufacturer (Figure 1-4).⁵

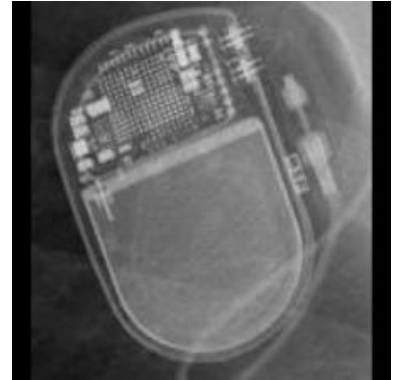
Unfortunately, this system has several shortcomings. First, it is time consuming, requiring multiple steps to be followed. Second, the most recently published CaRDIA-X version is over 9 years old, and so its performance is unknown on devices published since this date. Third, even at the time of publication, the study authors reported accuracies of only 90% in physicians using the CaRDIA-X algorithm.⁵



Medtronic Claria



Medtronic Advisa



Biotronik Evia

Figure 1-3 - Three CIEDs seen on chest X-ray. The left, centre and right images depict Medtronic Claria, Medtronic Advisa, and Biotronik Evia devices, respectively. The centre and right images are visually more similar, however, despite being of different manufacturers.

1.1.4 My proposal for a neural network approach to CIED identification

There are several aspects to the problem of CIED identification which may lend itself to a machine learning approach with convolutional neural networks.

First, X-rays are routinely performed following the implantation of most CIEDs, to document both lead position and the absence of post-procedure complications.⁶ Neural networks typically need to train from large datasets, comprising many examples (Section 2.8 - Training neural networks). Such examples are available in large quantities for CIEDs.

Second, CIED identification is a classification task with objective categories. A challenge that can sometimes be faced in training neural networks is the difficulty in establishing the gold standard correct answer. Examples of that challenge will be seen in later chapters. However, in this dataset of post-CIED implantation chest X-rays, we know the exact model of CIED present from the medical record of the implant procedure.

Third, heterogeneity within each class is relatively small. Whilst the X-ray appearances of a certain CIED category will vary across patients due to variable orientations of the device within the pocket and each patient's chest wall anatomy, the device itself is identical across all patients. This is in contrast to say a dataset of pictures of dogs, where dogs of different breeds and ages can appear markedly different.

Fourth, X-rays of CIEDs are relatively spatially invariant. This means that the X-ray appearance of two devices, one placed with the front surface of the device facing away from the patient's thoracic cavity, and the other device 'flipped over' with the front surface facing towards the patient's thoracic cavity, will appear as "mirror images" but otherwise visually indistinguishable from each other. This is a consequence of the devices being relatively thin (and so having no measurable depth that would cause ventral components appear larger in AP films, and smaller in PA films). This has the consequence that the X-ray appearances of devices in multiple

orientations can be simulated through a process of “data augmentation” (Chapter 2.9), allowing the neural network to generalise better to orientations not seen in the training dataset.

For these reasons, I decided to explore whether convolutional neural networks could be used to identify CIEDs from chest X-rays. This work is outlined in Chapter 4.

1.2 Identification of successful His Bundle Pacing

This section serves as a preamble to Chapter 5 - Identification of successful His Bundle Pacing using Convolutional Neural Networks.

1.2.1 The rationale for His bundle pacing

Cardiac pacing is a process by which the heart is electrically stimulated by an artificial pacemaker to trigger a heartbeat.⁷ This is traditionally done through a pacemaker wire inserted through the tricuspid valve and into the right ventricle. This process, termed right ventricular pacing, is effective at treating bradycardias, but results in a unphysiological depolarisation pattern within the ventricles, as the electricity spreads slowly through the right ventricular myocardium towards the left ventricle, rather than through the native conduction system.⁸ This slow depolarisation process manifests as a left bundle branch block pattern on the ECG, but it also appears to have a detrimental effect on cardiac function in some patients. Indeed, long-term right ventricular pacing has been found associated with ventricular impairment, heart failure and increased mortality.⁹

His bundle pacing is a novel form of pacing which aims to address this issue. His bundle pacing is typically performed by screwing a pacing wire, placed in the right atrium, into the interatrial septum, so that the native conduction system (the His bundle) can be depolarised directly.¹⁰ In many cases, a His bundle paced ECG can appear similar to a patient's intrinsic ECG, and non-invasive ECG mapping indicates that the depolarisation patterns from His bundle pacing are more physiological than those from other pacing methods.¹¹

1.2.2 His bundle pacing has a high barrier to entry due to the need for skilled ECG interpretation

The process of His bundle pacing is not hugely dissimilar from other forms of pacing, with a transvenous pacing lead inserted into the right atrium. From here, it is screwed into the right atrial septum so that it is able to electrically stimulate the His bundle. However, skill is required by the implanting team so that when pacing is performed, they can identify whether they have achieved true His bundle pacing, or merely traditional 'myocardial' capture.

There are broadly three types of cardiac pacing, or 'capture', that can be observed during His bundle pacing procedures.¹² These will be discussed in more detail in section 5.2.2, but they are (1) selective His bundle pacing, where the His bundle is directly stimulated by the pacing wire, without myocardial capture; (2) non-selective His bundle pacing, where the pacing wire stimulates both the His bundle and local myocardium; and (3) myocardial only capture, where there is a failure of His bundle pacing and only local myocardium is depolarised.

It is important that physicians performing His bundle pacing are able to discriminate between these responses. Discriminating between non-selective His bundle pacing and myocardial-only capture are particularly difficult; during both of these there is a slurred QRS appearance on the ECG immediately following the pacing spike, but only in the former has successful His bundle pacing been achieved. Failure to discriminate between these two would result in procedures being prolonged when His bundle pacing has already been achieved (in the case of non-selective His bundle pacing being incorrectly classified as myocardial only capture), or even more concerningly, patients being accidentally left with grossly unphysiological septal pacing when the team believe the procedure has been a success (if myocardial capture is incorrectly classified as non-selective His bundle pacing).

There are currently guidelines which assist physicians in discriminating between the electrophysiological responses to His bundle pacing. However, these are complicated and require several steps, including repeated comparisons between baseline and paced ECGs, and sometimes complex pacing manoeuvres to be performed using a pacemaker programmer.

1.2.3 Convolutional neural networks are a possible solution

A real-time system which is able to discriminate between the different ECG responses to His bundle pacing might reduce the barrier to entry for the procedure, procedure times, and morbidity. I have previous experience using convolutional neural networks to classify biological waveform data in the form of aortic pressure traces,¹³ and wished to see whether such an approach could be used to discriminate between 12 lead ECGs from His bundle pacing procedures showing selective His bundle pacing, non-selective His bundle pacing, and myocardial only capture. I explore this in Chapter 5.

1.3 Echocardiogram view identification

This section is a preamble to Chapter 6 - Echocardiogram view identification.

1.3.1 Echocardiography and difficulties for deep learning

Echocardiography is the most common form of imaging within the discipline of cardiology, with over 10 million studies performed per year.¹⁴ This is due to its unique combination of portability, availability and affordability. However, the provision of echocardiography is currently limited by the availability of the skilled workforce to perform and interpret these studies.^{15,16}

Processes to automate echocardiogram interpretation are therefore attractive. And yet, because of some inherent difficulties of the modality, remarkably little progress has been made in automated interpretation of these studies compared with other modalities such as magnetic resonance imaging (MRI)^{17–20} and computed tomography (CT)^{21–24}. There are many reasons for this, including:

1. Homogeneity of colour and brightness. In many applications of neural networks, colour provides useful information about the identity of an object. In echocardiography, however, all structures are grayscale. Furthermore, two neighbouring structures may even have the same *average* brightness, but only be distinguishable by a change in texture, which is difficult to quantify.
2. A lack of distinct borders. The nature of echocardiography means that acquired images are made up of speckles of varying density and brightness – a ‘perfect’ echocardiogram will still acquire variable texture within a structure such as the myocardium, which

presents as speckling. It is therefore rare to observe a definitive boundary between two interfaces when one examines at the pixel level, and so delineating structures may be more difficult.

3. Various forms of artefact. Whilst the above issues are inherent to the modality of ultrasound, there is a variable amount of artefact seen across studies. Artefact can be due to the inability of ultrasound to transmit across air filled cavities such as lung and dense structures such as ribs, which causes acoustic shadowing. There are also other idiosyncratic artefacts that are a consequence of the behaviour of sound waves (reverberation artefact, refraction artefact) and the way the incident beam is transmitted and processed (side lobe artefact, beam width artefact).
4. Paucity of meta-data to inform on the structures depicted in the image. In MRI and CT scans, the computer files produced by the scanner contain important information such as the angle at which the current picture is being taken and the name of the sequence. This can be used by automated image interpretation software to help identify what structures are likely to be present in an image and guide their analysis. Echocardiography files, however, are unable to store such information, as the position the sonographer is holding the probe in, and the order of acquisition, is unknown to the machine.

The above challenges mean that echocardiography has been a particularly difficult field for automated image interpretation.

1.3.2 Echocardiography ‘views’

Any automated echocardiography interpretation pipeline must first be able to recognise the correct video to analyse. This is because each echocardiogram study comprises upwards of 50 video loops providing depictions of slices of the heart in a variety of anatomical planes and orientations, also known as ‘views’. Therefore, before a video loop can be analysed, the view it represents must be correctly identified.

Several studies have previously applied neural networks to this task, but with variable results.^{25,26} One recent study using CNNs reported accuracies of only 84%,²⁶ noting particular difficulties in distinguishing between several different views that are taken from the apex of the heart with different probe orientations. Failing to distinguish these views leads to erroneous identification of the cardiac segments, which are an important basis for assessing myocardial infarction or ischaemia. If further automated analysis is to be done after view classification, it would rely on the CNN being highly accurate in selecting the correct video.

1.3.3 Current deep learning approaches for echocardiogram view identification

Efforts to date have shared a common approach of extracting a series of frames from echocardiographic videos and then classifying each of these frames in isolation as one of a series of views.^{25,26} The modal answer across a video is then chosen as the neural network’s prediction. This ‘frame-by-frame’ approach is straightforward but misses the opportunity to use temporal information of how features such as heart valves or ventricular walls move during the cardiac cycle.

There are several possible ways that this temporal information could be captured by a CNN, which can be simplified in the following manners:

1. Train a neural network to identify the view by learning how the activation patterns inside a standard frame-by-frame neural network evolve over the cardiac cycle.
2. Train a neural network to analyse the whole video directly, in one analysis. Such a ‘three-dimensional’ convolutional neural network would analyse data across frames as well as within.
3. Train a neural network to analyse two inputs: the standard frame-by-frame data and also ‘optical flow’ data. Optical flow data describes how structures move between successive frames. Such a network is therefore able to integrate both temporal and spatial signals.

In Chapter 6 of my thesis I explore all of these approaches and compare them with frame-by-frame approaches using several neural networks which are state of the art in other fields. I also assess the agreement between two human experts performing the same task.

1.4 Automated analysis of cardiovascular magnetic resonance anatomy images

This section is a preamble to Chapter 7 - Automated analysis of cardiovascular magnetic resonance anatomy images.

1.4.1 Rationale for an “early warning system”

Cardiovascular magnetic resonance (CMR) scanning is increasingly becoming the gold standard for assessing cardiac morphology and function.²⁷ One of CMR’s strengths is the number of different sequences which can be acquired during a scan. Separate protocols exist for acute myocardial infarction, stress testing, angiography, non-ischaemic cardiomyopathies, arrhythmogenic cardiomyopathies, congenital heart disease, pericardial disease, cardiac masses and more.²⁸ Furthermore, the different sequences making up these protocols can be acquired in a theoretically infinite number of scan planes at different slice thicknesses. For this reason, which sequences are going to be acquired during a scan are decided based on the patient’s clinical details and the clinical question, a process termed ‘protocolling’.

The difficulty with this strategy of protocolling, however, is that it cannot cater for unexpected pathology. For example, a patient undergoing a CMR for family screening for a cardiomyopathy may undergo a relatively focused scan to characterise the myocardium. However, if the aortic root appears dilated in these images (typically an asymptomatic and unexpected finding), current guidelines recommend the patient undergoes several further dedicated sequences.²⁸ It is therefore important for the scanning staff to identify the aortic root dilatation during the scan so these sequences can be performed before the patient goes home, preventing the need to call the patient back for a second scan.

A further complicating factor are contrast agents. Gadolinium is one such agent used in CMR scans. It works to shorten the T_1 and T_2 relaxation times of tissues, making tissues appear brighter on T_1 imaging in proportion to how much gadolinium is present. This allows CMR scans to ‘highlight’ areas of increased perfusion and/or poor venous drainage. However, some protocols rely on sequences to be performed before any contrast is given. This means that if an unexpected finding is identified only after the contrast has been administered, a patient may need to be brought back on another day to gather further pre-contrast images.

There is therefore a clinical need for a system which can identify unexpected CMR findings as early as possible during a scan. Radiographers can then react to these findings and amend the protocol as necessary. This could result in improved patient experience by reducing patient recalls, and improvements in the diagnostic quality of scans.

1.4.2 The axial anatomy sequence

A cardiac MRI scan begins with a series of brief sequences, termed “localisers”, which are used to orientate the scanning planes of the rest of the term. Following the localisers, the “axial anatomy” sequence is typically acquired. This comprises a series of slices which are axially-aligned on the entire patient, rather than aligned with cardiac landmarks. The slices typically range between the lung apices and upper abdomen, and are between 6 and 12mm in thickness. This sequence is particularly important when assessing for extra-cardiac pathology,²⁹ but it also provides the scanning team with the earliest multi-slice cross-sectional images through a patient’s heart, allowing rapid review of the gross cardiac anatomy, aorta, lung fields and mediastinum.

Although the anatomy sequence in principle contains useful information about heart structures, a human observer would have great difficulty interpreting them, because they are not aligned in the traditional cardiac planes (Figure 1-5).

In chapter 7 I explore a neural network that can make useful assessments from these early sequences, to provide timely notification to staff of abnormalities that they would only be able to recognise after later views. This could allow earlier amendment of scanning protocols and increased efficiency.

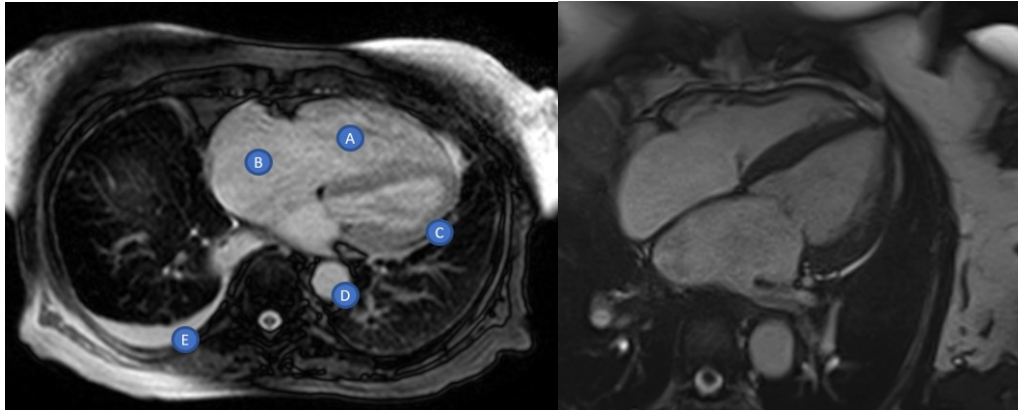


Figure 1-5 – LEFT: A slice from an axial anatomy planning sequence. In this slice the heart is visible, and it appears the right ventricle (A) and right atrium (B) are dilated. However, measurement of cardiac structures is not advised in these non-standard views, because the cardiac structures are not appropriately aligned with the scan planes, and the contrast between the left ventricular wall and blood pool is poor (C) compared with a dedicated 4 chamber sequence. In this picture, further diagnostic information is available, including imaging of the descending aorta (D) and evidence of a pleural effusion (E). RIGHT: An example image of a dedicated 4 chamber acquisition. The contrast between the left ventricular wall and blood pool is noticeably improved, and the four cardiac chambers are correctly imaged using an appropriate scanning plane, or “on axis”.

1.5 Aims of this thesis

Whilst computers have become a key component of modern medicine, computer-assisted interpretation of investigations remains out of reach in most settings, especially in the field of medical imaging.

The main hypothesis of this thesis is that deep learning using convolutional neural networks can provide fast and accurate automated processing of cardiology investigations, including X-rays, echocardiography, cardiac magnetic resonance imaging and electrocardiography. Within this body of work, I will specifically investigate the performance of convolutional neural networks in four tasks:

1. Can a neural network identify the exact model of a pacemaker or defibrillator a patient has from a chest X-ray, and how does its performance compare with that of a cardiologist using a manual algorithm?
2. Can a neural network discriminate between the different electrophysiological responses to His bundle pacing?
3. Can a neural network accurately identify the anatomical structures present in an echocardiographic video (i.e., the echocardiographic ‘view’)? Can current state-of-the-art approaches be improved upon by creating neural networks which can process the temporal data present within videos, rather than just the spatial information from individual frames?
4. Can a neural network be used to process the axial anatomy sequence images acquired at the earliest stages of a cardiac magnetic resonance imaging scan, and how do the measurements made by this network compare with the measurements made by a physician using the information from a full scan?

2 Methods - Key machine learning & deep learning methodology

In this chapter I will introduce the important aspects of machine learning and, in particular, convolutional neural networks. Study-specific methods will be introduced in the results chapters (sections 4.3, 5.3, 6.3 and 7.3).

2.1 An introduction to machine learning

Machine learning is a revolution in computer science that will have major consequences for the practice of cardiology.¹ Although the detail may be intimidating, the principles can be seen as an extension of widely familiar techniques, and a new workflow. Historically, a computing task is programmed by a human converting a mental understanding of the required steps into code. Instead, in most cases of machine learning, the human provides the computer a bank of questions and correct answers, along with a broad architecture for building an automated solution. The computer then progressively develops a solution.

What catapulted machine learning into being orders of magnitude more effective was the discovery that numerous sequential layers of simple arithmetic are surprisingly powerful at solving previously difficult problems. This “deep learning” design has been effective in image interpretation, particularly with the convolutional neural network (CNN) design that is inspired by the human visual cortex.³⁰

Within cardiology, CNNs have proven highly effective in processing clinical data that is 1-dimensional (ECGs, cardiac pressure waveforms)^{13,31}, 2-dimensional (chest X-rays, echocardiography and MRI stills)^{25,32} and 3-dimensional (cardiac MRI and echocardiogram

videos, cardiac CT volumes).³³ In some cases the CNNs' performances may surpass those of cardiologists.

Machine learning will play an increasing role in medical practice and, as with any diagnostic test or piece of medical equipment, an understanding of these systems will better equip medical staff to interpret these systems' results.

2.2 Simple forms of machine learning

Many are surprised on realising that many forms of machine learning are fundamentally simple, and indeed a decade ago would have been called “statistics”. As a simple example, in cardiology we have a formula “220 – age” to predict a patient's peak heart rate on exercise.³⁴ This formula arose from simplifying a simple linear regression:

$$\text{predicted heart rate} = \text{constant} - (\text{another constant} \times \text{age})$$

Originally, pairs of ages and heart rates were used to find the best constants, and these were then rounded to 220 and 1, respectively.

Cardiology has many examples that are one step more complex, in the form of risk scores. Here, what is predicted is not a measurement, but whether the patient will (for example) live or die. It is never possible, of course, to predict this with certainty and therefore what is produced is a probability. However, this can nevertheless be very informative. In Figure 2-1 is an illustration (using simulated data) of the risk model for age and cholesterol predicting death over the next 10 years, which has several interesting features. It is curved, because it must be; at the age of 40, the risk of death is so low that even big differences cholesterol results in only a small increase in

death. At the other extreme, for a 90 year old the risk of death over the next 10 years is so high that the patient's cholesterol has little bearing on it. It is in the middle zone that cholesterol has the highest influence on the probability of death.

These familiar practices in clinical cardiology are at the simple end of the same spectrum of machine learning. We can start with a dataset of a continuous variable input (e.g. age) and the corresponding output (peak exercise heart rate), and we are familiar with a computer calculating the best fit relationship between them, which we call **linear regression**. This can be extended to outputs (e.g. death) that are probabilities (**logistic regression**).

A key insight from these familiar regression tasks is that while the process of calculating the best fit relationship is time-consuming (typically requiring a computer), the process of applying that relationship once developed is relatively easy, e.g. "220 minus age".

When developing a regression model, the researcher supplies the computer with a series of example cases and what type of line of best fit we want it to draw: straight lines or particular classes of curves.

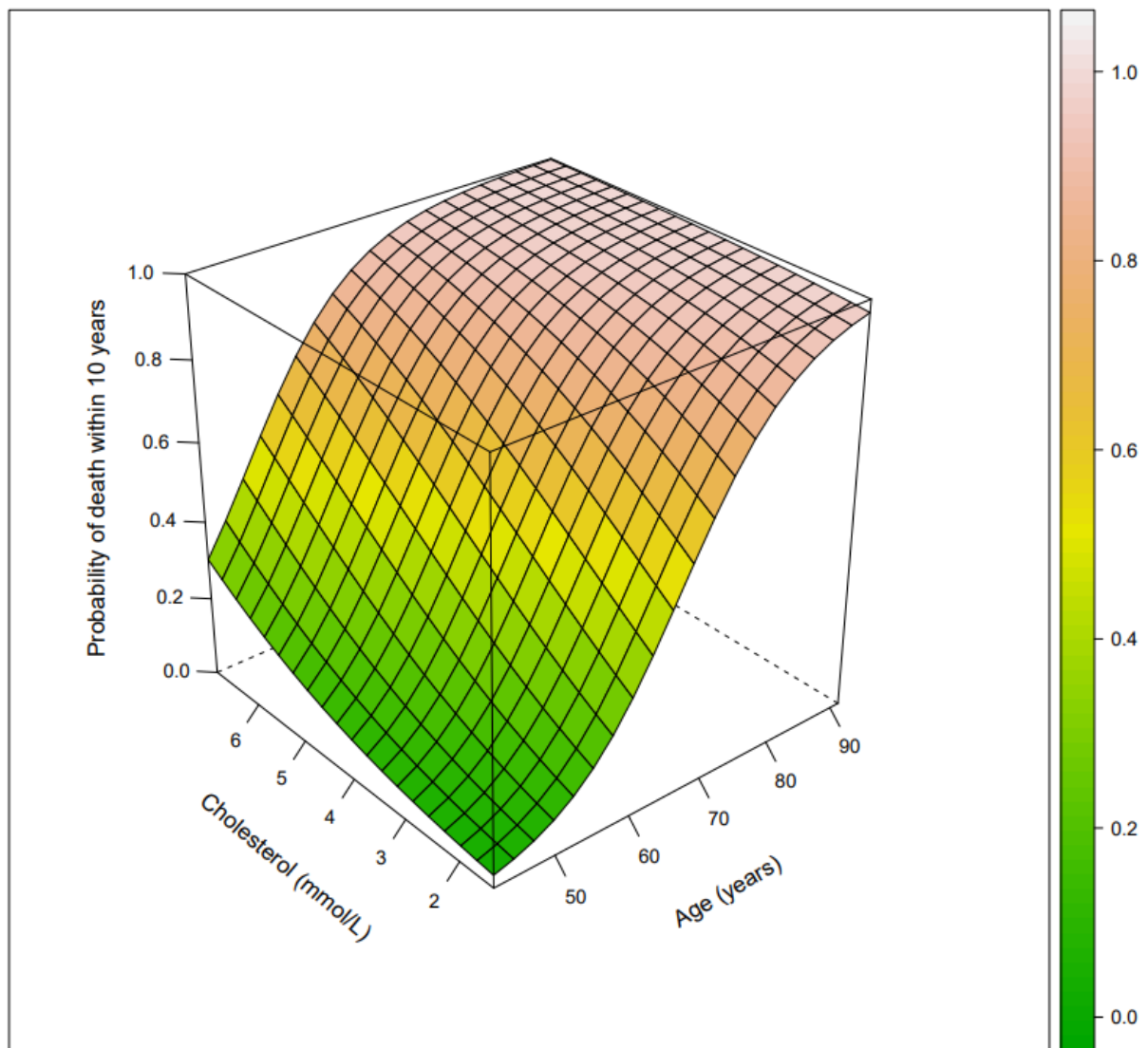


Figure 2-1 - The relationship between age, cholesterol and mortality (probability of death).

2.3 Supervised and unsupervised learning

For a computer to fit a regression model for us, we must supply it with example pairs comprising the predictor (e.g. age) and the variable we are trying to predict (e.g. maximum heart rate). This process is termed “supervised learning”, because the data we are providing includes the correct answers.

There is another type of task that machine learning can perform, called “unsupervised learning”. There is no specific correct answer, but we ask the computer to discover patterns in the data. For example, it could be finding phenotypic variants within a large group of patients who superficially appear to have one condition.³⁵

Both supervised and unsupervised learning have undergone a revolution in recent years, for two reasons. First, algorithms have been developed to tune even models that are millions of times more complex than conventional regression models.³⁶ Second, computers have become fast enough to perform the tuning steps which are vastly increased not only in number but also in subtlety (because the interactions between the different elements are unimaginably more numerous).³⁷

2.4 Neural networks

Neural networks are assembled from large numbers of simple elements in an arrangement inspired by the biological networks of neurons in the animal brain.³⁰

Akin to the animal brain, these neural networks typically comprise a series of layers of individual units termed ‘neurons’. Like biological neurons, these neurons receive inputs from other neurons and combine these inputs in some way. Just as a biological neuron’s output is its frequency of firing, these computational neurons output a number.

In a sense, a conventional statistical model can be reimagined as a neural network. Inputs contribute either positively or negatively to the output of the model, which may be a risk score or a probability, e.g. of death. Figure 2-2 shows an example neural network of this type.

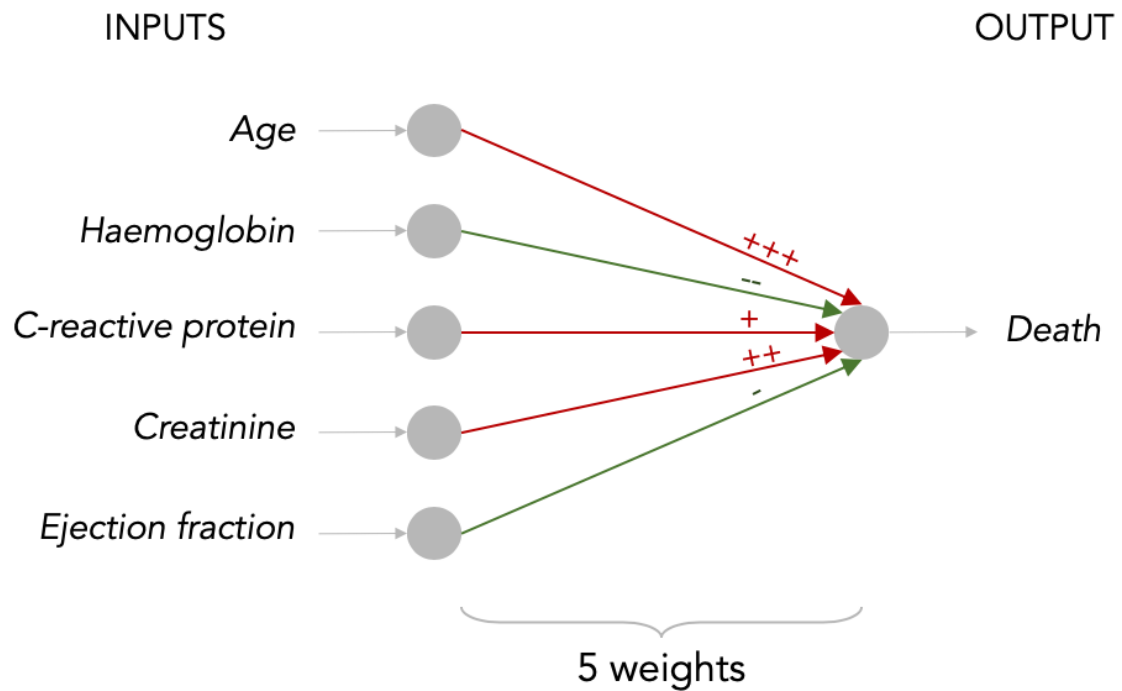


Figure 2-2 - The simplest form of neural network. 5 input neurons, corresponding to 5 input variables feed into 1 output neuron, corresponding to the chance of death. The input neurons either increase (age, c-reactive protein, creatinine) or decrease (haemoglobin, ejection fraction) the activity of the output neuron, whose final value can be measured to provide a probability. The connections (weights) between the input neurons and the output neuron are adjusted to give the most accurate answer.

Neural networks can extend this concept by having intermediate calculations between the input and output, however. These intermediate values are described as “hidden” because the end user does not see them. These intermediate steps allow the network to compute relationships that are more complex than simple straight-line dependencies (like heart rate = 220 minus age). Each intermediate step can be very simple (such as addition or multiplication) but combining several one after the other can produce surprisingly sophisticated processing.

In the neural network depicting in Figure 2-3, termed a “multi-layer perceptron” (MLP), there are two hidden layers. Each neuron in the first hidden layer receives only the raw data and can therefore only compute simple linear functions of them, such as $220 - \text{age}$, or $(\text{haemoglobin} \times 7) + (\text{troponin} \div 42)$. In the second hidden layer, however, each neuron has access to the outputs of the first hidden layer, and therefore can compute more complex relationships by combining them. The output can therefore have a much more sophisticated dependency on the input values than can a simple linear model depicted in Figure 2-2. Modern neural networks can have dozens of layers.

We store the strength of each of the synapses in a network as a number, which we term a ‘weight’. A positive weight can be thought of as a stimulatory synapse, and a negative weight as an inhibitory synapse. By modelling the network this way, we can perform a series of 3 simple arithmetic operations (one for each ‘layer’ of the network) to translate the 5 input numbers into an output number.

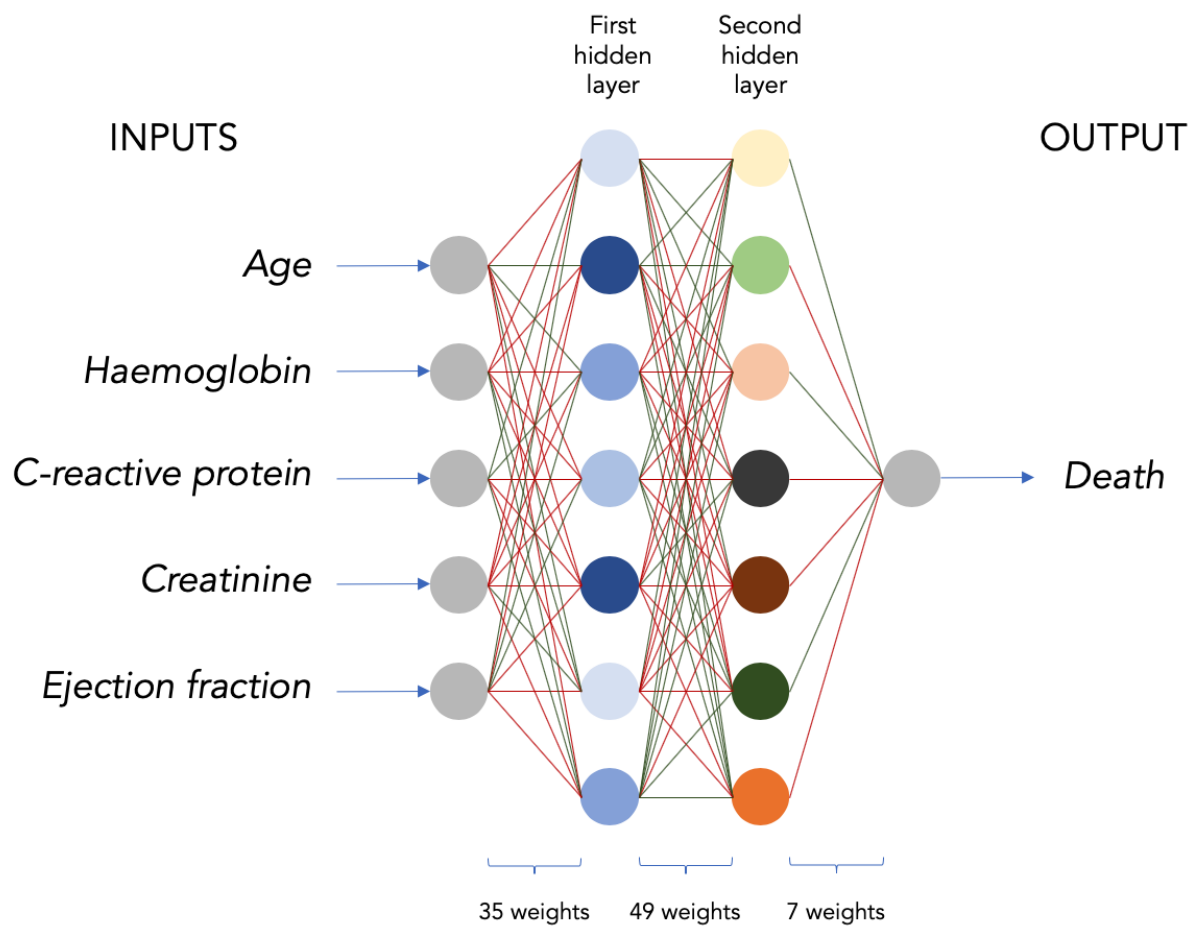


Figure 2-3 - An example “classical” neural network, also termed a multi-layer perceptron (MLP). It is made up of a combination of stimulatory (green) and inhibitory (red) synapses, which process a series of inputs (age, haemoglobin level, c-reactive protein level, creatinine level and ejection fraction) to yield a probability of death. The final value, and therefore decision of the network, is decided by the relative strength of the synapses between the neurons (termed parameters, or ‘weights’).

When this network is created, the weights are set randomly and so the initial data output by the network is meaningless. However, these weights are then adjusted through “training”. Training is process by which we repeatedly show the network many sets of input data, along with the correct answer (a form of supervised learning). The network then adjusts each weight in the direction that would have yielded a better (more correct) answer. The network therefore eventually “learns” the best way of processing the 5 baseline characteristics, in our example, to make the correct prediction for that patient.

At first it seems spectacularly unlikely that an automated process could make millions of small adjustments result in a meaningful neural network evolving. However, that was the stunning insight behind “gradient descent”, the mathematical process of adjusting these weights.³⁶ When there are very large numbers of these weights, surprisingly, it becomes increasingly likely that the gradient descent algorithm will gradually move towards a more successful network.

2.4.1 Advantages of neural networks over classical approaches such as logistic regression

In the above example, it might not be clear why an approach using a neural network such as an MLP is any more advantageous than fitting a logistic regression model where we try to predict the odds of dying using the predictor variables. In fact, if the number of hidden layers in the MLP was 0, and we just connected the input neurons directly to the output neurons with adjustable weights, our approach would be identical to regression. However, the inclusion of the hidden layers allows the MLP to perform “deeper” processing of the data, in two specific ways:

- The network can create complex *non-linearity* by assembling a pipeline of simple standard units

- Signals from different sources can interact at *multiple stages*

Although these two features seem simple, their combination is extraordinarily powerful, as explained below.

2.4.2 Origin and utility of non-linearity

After each neuron, we use a simple mathematical function to transform the output.

Surprisingly, a function termed the “rectified linear unit” (ReLU) has turned out to be a good choice.³⁸ The output values a ReLU produces for input values between -5 and +5 are shown in Figure 2-4.

ReLU merely changes negative numbers to 0, whilst letting positive numbers pass through unaffected. However, because the neurons and their associated ReLUs are cascaded in large numbers, organised in several layers, they can effectively emulate many complex mathematical functions.

For example, by combining just two neurons in a layer between an input and an output, with a ReLU immediately following each neuron, we can mimic more complex functions such as converting a number to its absolute value (removing any negative signs; Figure 2-5, yellow graph) or approximate something close to a sigmoid function (Figure 2-5, blue graph).

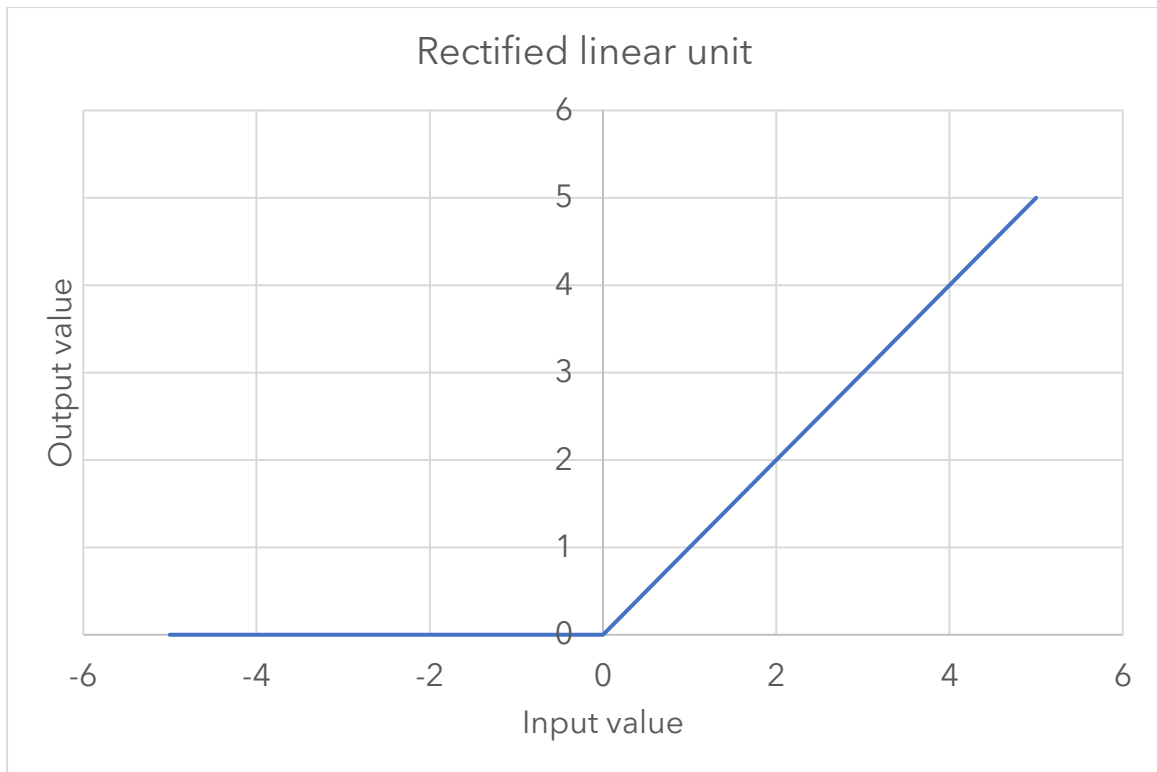


Figure 2-4 - The rectified linear unit (ReLU) activation function. Values below 0 are set to 0; positive values are unchanged.

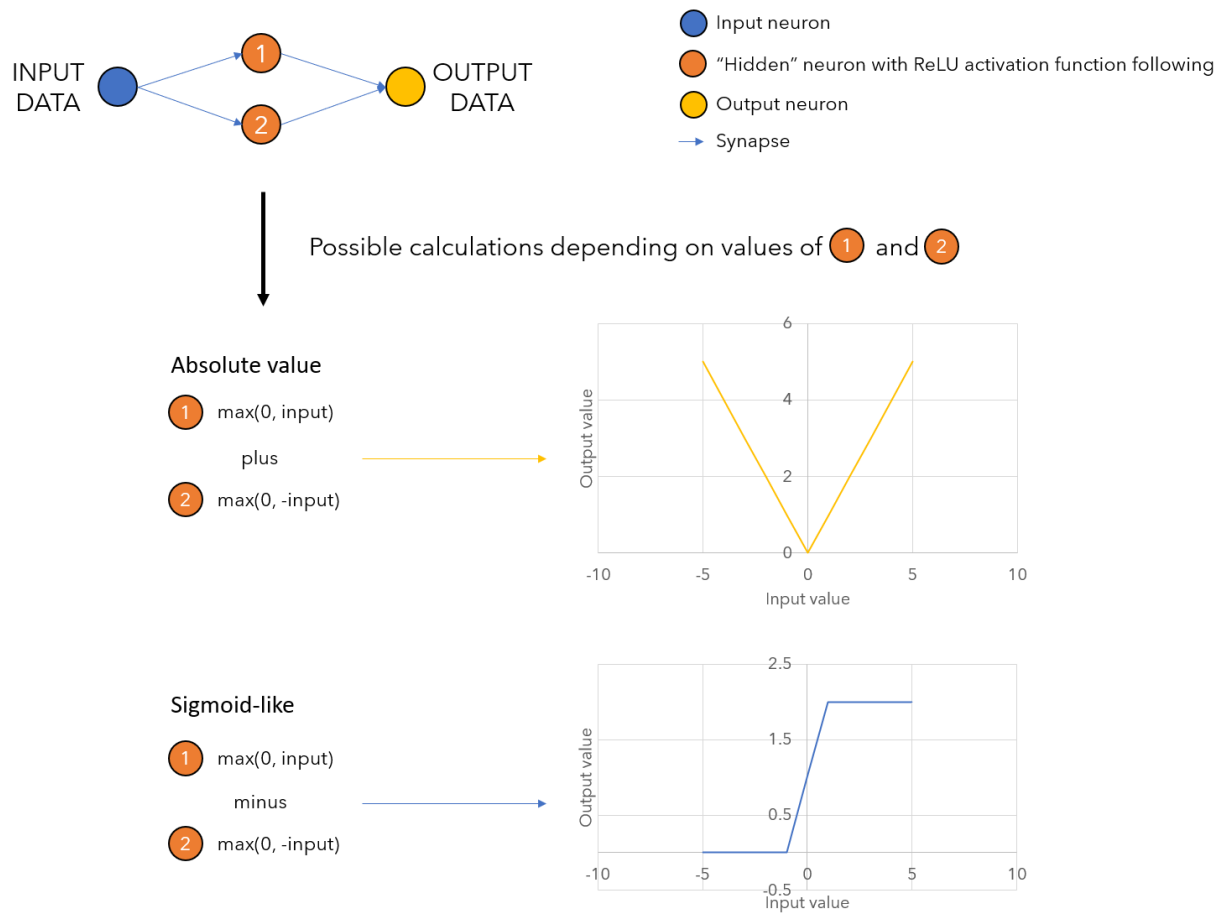


Figure 2-5 - A schematic of a simple neural network with two hidden units between an input and output neuron. The hidden units each comprise a neuron and a rectified linear unit (ReLU). These take a maximum of 0 and an input value, depicted as " $\max(0, \text{input})$ ". With just two neurons, we can create a small network which calculates the absolute value of an input value (yellow graph), or something which approximates a sigmoid function (blue graph).

2.4.3 More layers allow inputs to be combined at different levels

In conventional algebra, a function z can depend on x and y in various different ways. There is a family of relationships within which x and y do not interact, namely, $z(x, y) = f(x) + g(y)$. In other words, x and y are processed separately and the results are added at the end. An example of this is the Glasgow Coma Scale, where the responses of the eyes, motor system and voice are assessed separately and summed.

There is another family of relationships in which x and y are combined by an initial linear function, and then acted on by a non-linear function. For example, calculation of body mass index (BMI) from Imperial units:

$BMI(stones, pounds, feet, inches) =$

$$(stones \times 14 + pounds) \times 0.453$$

$\div$

$$((feet \times 12 + inches) \times 0.025)^2$$

Note that in this formula, the stone and pounds interact together linearly, and the feet and inches interact together linearly. The two results then interact non-linearly, and there are no other interactions. This process could *only* be modelled using multiple layers.

The family of functions we can model with numerous layers is vastly richer than the above. It allows each input variable to be transformed non-linearly, combined non-linearly with other variables, and the results further processed non-linearly through multiple steps.

Such complex functions can be “learned” in the hidden layers, and for this reason, high performing ‘deep learning’ network typically have dozens of layers.

2.4.4 Multi-layer perceptrons in cardiology, and their limitations

Whilst MLPs have been used in multiple settings in cardiology,^{39,40} they have not been adopted as substantially as some other network designs, especially in the field of automated medical image processing.

A major reason for their limited application is their poor “scaling”; as the number of inputs into the network increases, the number of weights the network must store increases dramatically.

For example, one high profile paper used a neural network with 48 layers to process images of skin lesions and showed dermatologist-level performance in identifying cancers.⁴¹ The images they fed into their neural network were 299 pixels in height, and 299 pixels in width, meaning 59,800 pixels in size. To feed such an image into an MLP, we would therefore need 59,800 input neurons. Just to connect this layer to the **first** hidden layer of 100 neurons would require almost **6 million** weights, requiring large amounts of training data and time.

Furthermore, if that picture is shifted rightwards by one pixel, the inputs to every neuron in the network completely change, and it will no longer be able to recognise the image.

2.5 Convolutional Neural Networks

The insight that catapulted deep learning into the forefront of image analysis was that the same visual feature might appear in any one of hundreds of positions on a large image, and should ideally be recognisable by the network, regardless of the position. To achieve this, why not have the neural network view each part of the image in turn? That way, the network learns to recognise important features, rather than their arbitrary positions within an image.

This is efficient, because, even if the image is tens of thousands of pixels in size, a network could be set to view many areas, each of which is much smaller, thereby needing many fewer weights in the network's design.

However, simply reducing the area inspected is not sufficient because that would not allow the network to recognise features that are larger than each small area. That is where the second insight came in, from how the mammalian visual system solves this problem.

It was this approach that the skin lesion study authors successfully used: the convolutional neural network.³⁰

2.5.1 Inspiration from the mammalian brain

In the 1960s, Hubel and Wiesel found specific cells in the first layer of a cat's visual cortex (V1) which depolarised when the cat was viewing a scene containing bright lines of a certain orientation.⁴² Other neighbouring cells depolarised when lines of different orientations were present, and other cells depolarised in the presence of specific colours, or movements. Damage to these fundamental cortical layers leads to complete blindness. However, damage to other occipital lobe structures can lead to specific defects in higher-level image processing, such as prosopagnosia (the inability to recognise faces) in humans, following damage to the fusiform gyrus.⁴³

These findings underline the fundamental workings of both the mammalian visual system and CNNs: the identification an object involves earlier layers identifying basic visual features present in a scene, and later (deeper) layers combining these features to make an overall decision.

2.5.2 Convolutional neural networks learn by matching templates

The way CNNs do this is simple: template matching. Whilst classical neural networks such as MLPs learn how to process data from individual pixels, CNNs instead slide (or ‘convolve’) a large series of small templates, termed ‘kernels’, through an image and record how well each area matches.

Figure 2-6 shows an example convolutional neural network which aims to identify whether an image is a nought or a cross for a game of noughts and crosses. It comprises only two layers, though, in practice, convolutional neural networks have many more.

The first layer the example network is a “convolutional layer” which includes 4 small templates, termed “kernels”.

When we feed an image into the network in Figure 2-6, the images enter the convolutional layer. Then, each kernel in that layer slides, or “convolves”, through the image and records how well each area matches. Through each kernel, therefore, a new image is created, termed a “feature map”, which indicates how strongly each area of the original image matches that specific kernel.

When we feed an image of a hand-written circle (nought) into the convolutional layer (Figure 2-6, top row), there are different areas of the image which match each specific kernel; for example, the side of the circle matches the vertical kernel, whereas the top matches the horizontal kernel.

However, when we pass an image of a hand-written cross into the convolutional layer (Figure 2-6, bottom row), we find a different pattern: we find no matches for either the horizontal or vertical kernels (the feature maps are empty), but very strong matches for the diagonal kernels.

The second layer in the example network is the “fully connected” layer. This layer is fundamentally identical to those in a multi-layer perceptron discussed previously (Figure 2-3), in that it is merely a layer where every neuron in the layer before (the convolutional layer) is

connected to every neuron in the layer following (the output neurons). The fully connected layer therefore takes the results of the final convolutional layer in a network and translates this into a prediction. In our example, an image containing many matches from all 4 kernels is likely to be a nought. In contrast, an image containing only matches for the diagonal kernels is likely to be a cross. This has been “learnt” by the network by adjusting the weights of the synapses between the vertical and horizontal kernels to be inhibitors for the cross class and stimulating for the nought class.

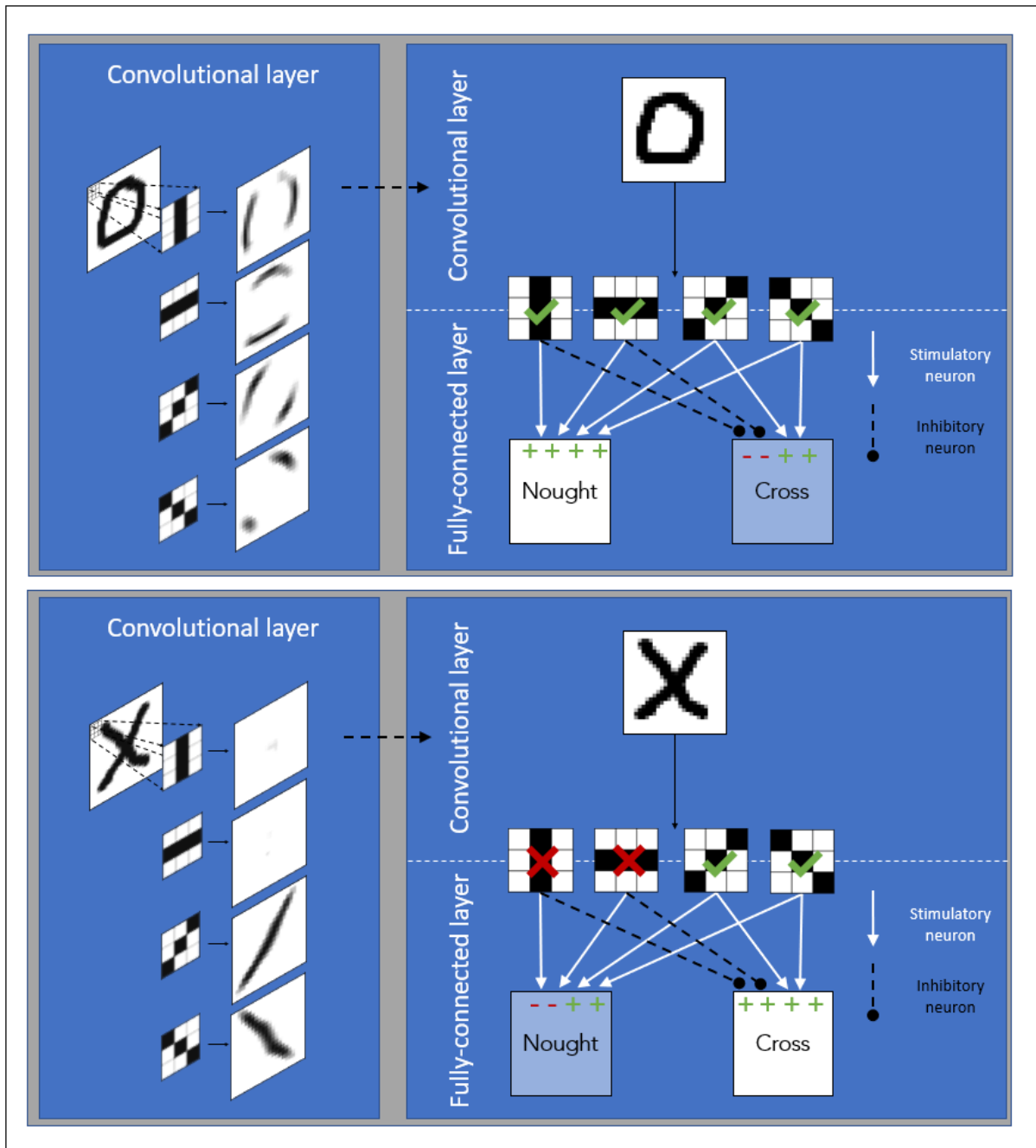


Figure 2-6 - A schematic of a two-layer neural network designed to identify whether an image is of a nought or a cross. In the top row, we can see the image of the nought contains areas which match each of the four kernels in the convolutional layer. The cross image (bottom row), however, doesn't contain any features that match the vertical or horizontal kernels. The fully connected layer allows the strength of these detections to be translated into predictions; the weights connecting the vertical and horizontal kernel matches to the 'nought' prediction are stimulatory, but they are inhibitory for the 'cross' class.

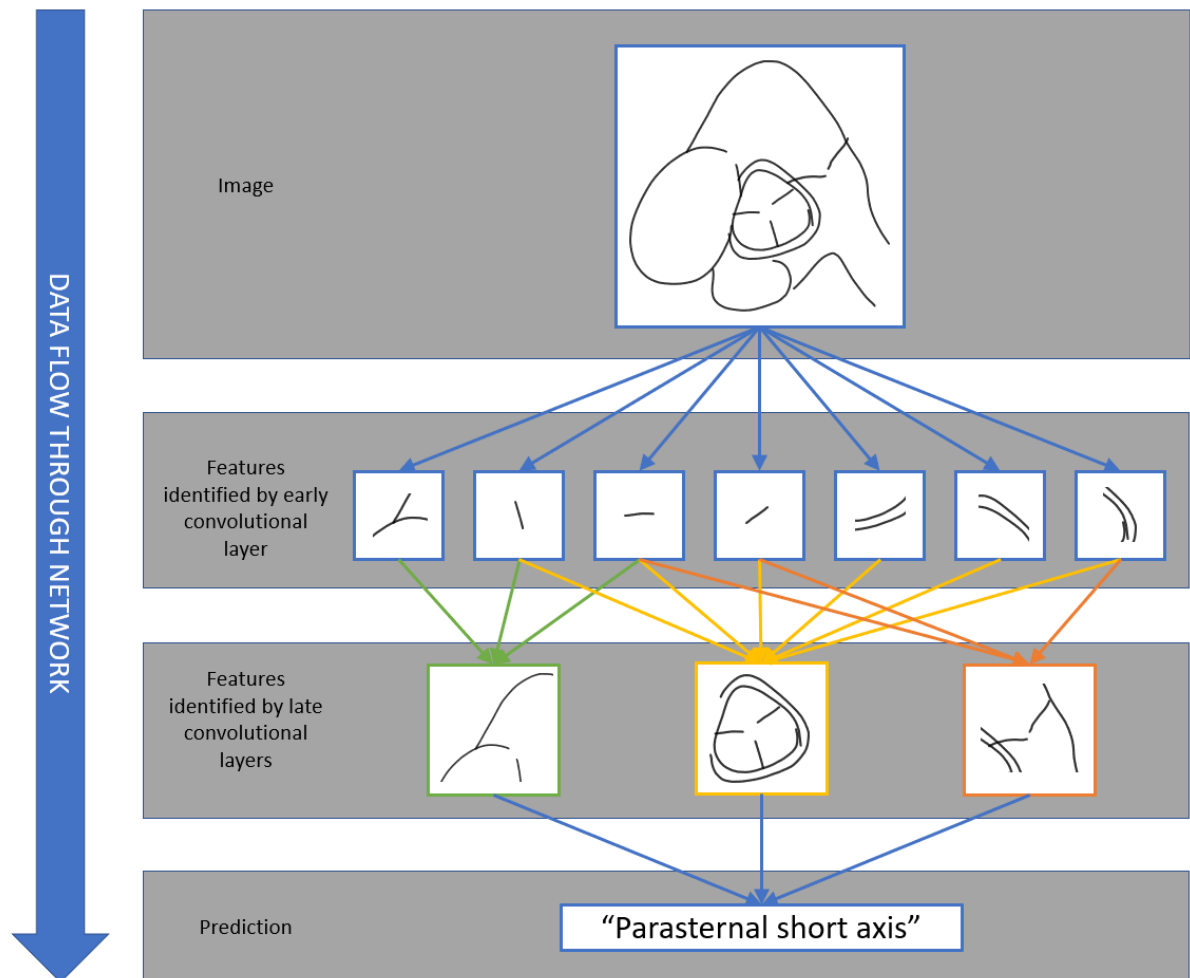


Figure 2-7 - A schematic of a neural network used to identify echocardiographic views. Early convolutional layers in this network identify basic features such as edges, whilst later layers combine these features to identify anatomical structures. The presence or absence of these structures leads to a final prediction by the network.

In practice, CNNs typically include several convolutional layers, with deeper layers convolving kernels through the feature maps produced by the preceding layers to identify more and more complex images. For example, a 3-layer CNN trained to identify echocardiogram views will use its early layers to identify relatively simple features within an image (Figure 2-7). Later layers in the network will combine these observed features to identify anatomical features which the neural network can use to make a decision on which echocardiogram view is being depicted.

During training, this convolutional network will learn the optimal weights in the fully-connected layer, but importantly it also **learns the optimal kernels** in the convolutional layer. This means that these small templates are not programmed manually, but rather naturally develop during the training process. This again is analogous to the mammalian brain; Hubel and Wiesel found that kittens which are raised in a world consisting of either solely vertical lines or solely horizontal lines end up being completely blind to objects of the opposite orientation when they are introduced into the real world,⁴⁴ presumably because they have not developed systems for recognising them.

2.5.3 Advantages of convolutional neural networks over classical neural networks

In the above example, the advantages of CNNs may not be clear. For example, the image could have merely been fed into a classical MLP for training, rather than a CNN. However, the CNN offers several distinct advantages:

1. **CNNs are efficient.** In our above example, if we store each kernel as 9 numbers (a 3 by 3 grid), the entire convolutional layer can be represented using $4 \times 9 = 36$ weights. We then require just 8 further weights to connect each of these 4 kernels to the two classes

in the fully connected layer, making a total of 44 weights. In contrast, if we used just a single-layer MLP as a network to process low resolution images of 28 by 28 pixels, we would need $28 \times 28 \times 2 = 1568$ weights. Generally, networks which can perform a task with fewer weights can be trained more quickly and with less data. Whilst in practice CNNs include multiple convolutional layers with many more kernels per layer, they remain far more efficient for processing images than MLPs.

2. **CNNs are (relatively) “spatially invariant”.** In our CNN example, we would expect the neural network to work even if the size of the nought or cross in the image changed; a cross confined to the left half of the image would still show strong matches for the diagonal kernels and little matches for the vertical and horizontal kernels. However, when using an MLP, moving an object within an image just one pixel leftwards results in every single pixel in the image feeding into a different input neuron, and a completely different set of activation patterns.
3. **A CNN trained for one task is a good starting point for beginning training for another task.** As shown above, CNNs work by identifying simple features present within an image and combining these to identify the contents of an image. The early layers of a CNN trained to perform one task can therefore be re-used in other tasks, skipping the need to re-learn the optimal kernels. This dramatically reduces training time and the amount of data required. This process of ‘transfer learning’ has been used extensively in medical imaging studies where a CNN has initially been trained on large generic datasets such as “ImageNet” before being ‘tuned’ on more specific medical datasets.^{32,41}

2.6 Convolutional neural networks beyond images

Because the CNNs above deal with two-dimensional images by sliding kernels across them in two dimensions (across the width and height of the image), we term them two-dimensional (2D) CNNs. However, one-dimensional (1D) and three-dimensional (3D) CNNs also exist and have shown great promise in analysis of non-image data.

1D CNNs are for analysing one-dimensional data, such as monitoring a variable over time. Such data are very frequent in cardiology, and include electrocardiograms (ECGs), blood pressure, coronary pressure measurements taken during fractional flow reserve (FFR) recordings,¹³ and photoplethysmography (PPG) traces from pulse-oximeters.⁴⁵

3D CNNs are able to analyse three-dimensional data. The most common form of this is video data, which can be thought of as having a height and width in pixels, but also a depth, corresponding to the number of frames a video contains. Individual frames within a video can be analysed using 2D CNNs, but by doing this we lose the ability to compare neighbouring frames for subtle changes such as small movements. A 3D CNN, however, allows tracking of structures between frames, as the 3D kernels in a 3D CNN convolve not only over the width and height of an image, but also over time, by convolving *through* the frames.³³

In this thesis, I will employ 1D (chapter 5), 2D (chapters 4, 6 and 7) and 3D (chapter 6) CNNs.

2.7 Convolutional neural networks' strengths extend beyond classification: regression and segmentation

In the above examples, I have shown how CNNs could be used in **classification** tasks, such as deciding whether a patient will die, whether an image is a nought or a cross, or what view is depicted in a cardiac MRI scan (Figure 2-8, left panel). However, CNNs can be used for other tasks, including **regression** and **segmentation** tasks.

In a regression task, the output of a neural network isn't a predicted class, but instead a continuous number. Such a number could be a predicted life expectancy or co-ordinates of a region of interest within an image. A common regression problem where CNNs are employed is in identifying a "bounding box" around an object present within an image (Figure 2-8, middle panel). This can be achieved by training a neural network with 4 output neurons: 2 neurons for outputting each of the row and column locations of the box, and 2 neurons for outputting the width and height of the box.⁴⁶

In contrast, in a segmentation task we want the neural network to provide us with an output image where each pixel is classified by what it corresponds to. For example, a neural network to segment the ventricles in a cardiac MRI image will hopefully identify the pixels corresponding to the left ventricle as one class, the right ventricle as another class (Figure 2-8, right panel). A segmentation problem can therefore be thought of as a classification problem, where each pixel is classified by what it contains.⁴⁷

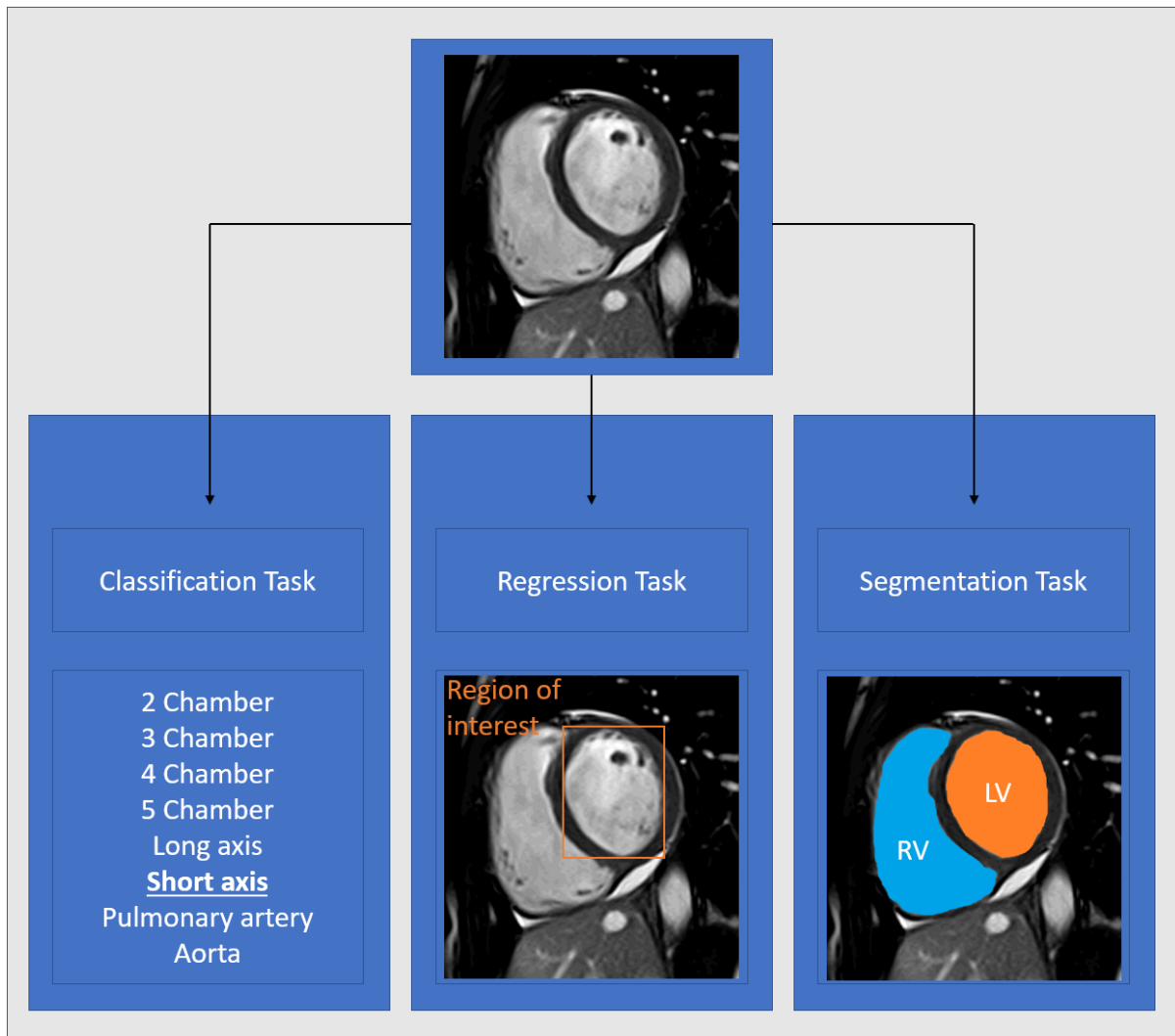


Figure 2-8 - CNNs can be used for classification, regression and segmentation problems. Classification problems may involve deciding which class an image corresponds to (left panel). Regression tasks involve predicting one or more continuous values from an image, which could be used, for example, to locate a bounding box (middle panel). In segmentation tasks, the neural network may classify each pixel within an image as one of several classes (right panel; LV = left ventricle, RV = right ventricle).

2.8 Training neural networks

We have explored the applications of CNNs for 1-, 2- and 3-dimensional data processing, in classification, regression and segmentation tasks. As mentioned in section 2.4, these networks must be ‘trained’ before they can be used.

However, to train the neural network, we require several things:

1. A series of **training data**, which comprises examples of the data the neural network is to learn to process, and their associated ‘labels’ - the correct answer the neural network should provide for that example.
2. A series of **validation data**, which are further examples of the data. The validation data are not used to train the network, but instead are used to gauge the neural network’s performance on data it has not seen before, in between the training steps, to assess progress. If progress is not good, it may be appropriate to change the entire design of the network. Because there is a risk of inadvertently tailoring the network design to match our particular validation dataset (Table 2-1), we do not report the ‘final result’ using this validation dataset.
3. A completely independent third series, called the **testing data**. Although the terminology is somewhat confusing, especially to those used to conventional medical statistical terminology, this is the standard in machine learning. The principle of the test dataset is that it is only tested once, with the final model, when ready for publication. Therefore, any accidental over-tuning of the network design to the validation dataset will give no

benefit on this final dataset. This should therefore be an unbiased estimate of a subsequent real-world performance (Table 2-1).

4. The number of **epochs** for which the neural network will be trained. An epoch refers to a full cycle of the training data. This means that the number of epochs refers to how long a neural network gets trained for; 5 epochs means it will see every example in the training dataset 5 times.
5. The **neural network** itself, which starts off randomly tuned before the training process begins. There are many different designs, or ‘architectures’ which can be used. Generally, those with more layers have greater learning capabilities, but are more difficult to train, requiring more time and data to do so. Furthermore, the patterns of dataflow within these networks can vary, making different network architectures suited to different tasks.
6. The **batch size**. This is how many images are considered together at one time (in a batch) when adjusting the weights of the network. A typical size is around 32. This means the mathematical algorithm views a batch of 32 images simultaneously and attempts to tune its several million weights to slightly improve its performance on this set.
7. The **loss function**. This is a measure of how wrong the network is on a particular batch. One could imagine this to be on a scale of 0 (the network functions perfectly on this

batch) to 1 (it gets everything wrong). In practice, we usually use a function that can become arbitrarily large, because they are more effective.

8. The **optimizer**. The optimizer is the program that adjusts the weights within the neural network to improve its accuracy. It does this after every batch. An important feature of many modern optimizers is that they keep a track of the previous adjustments they made to each weight. If the optimizer finds, for a particular weight, that it is alternating between increasing and decreasing it, it knows the ‘sweet spot’ is probably somewhere between, and it will make smaller adjustments. In contrast, if it finds itself consistently adjusting a certain weight in the same direction, it may become bolder and make larger adjustments. This phenomenon is termed ‘momentum’, and the exact behaviour of the optimizer can make a big difference to how quickly the network learns.
9. The **learning rate**. This is a starting value for how quickly the optimizer should adjust the network’s weights.

The training process is outlined in Figure 2-9.

Dataset	How often model is changed to fit this dataset	Tendency to over-estimate subsequent real-world accuracy
Training	Millions of times, directly by the fitting process	Very large, since the weights are deliberately tuned to maximise efficacy*
Validation	Several times, indirectly by humans changing structure of network in response to unsatisfactory results of training	Unknowable, but larger when human has performed many cycles of trial and error, or the validation dataset is small
Testing	Never	Zero**

Table 2-1 - How each of the training, validation and testing datasets influence the network, and how closely the network's performance on these datasets reflect its "real world" performance.

** A very deep neural network may have sufficient weights to merely 'memorise' the training set examples, without learning any useful general rules which would allow it to meaningfully classify novel data. This would manifest as a perfect training set accuracy, but very poor validation and testing set accuracy. This problem is termed "overfitting" and is analogous to overfitting in multiple regression in standard statistics, where there are an inappropriately large number of variables for the number of observations. I will discuss dealing with overfitting in a later section.*

*** This is assuming the test set is representative (unbiased) and large. If the testing dataset is highly unrepresentative of real-world data, then the performance of the network cannot be accuracy estimate from the testing dataset. However, it will still be the least biased estimate.*

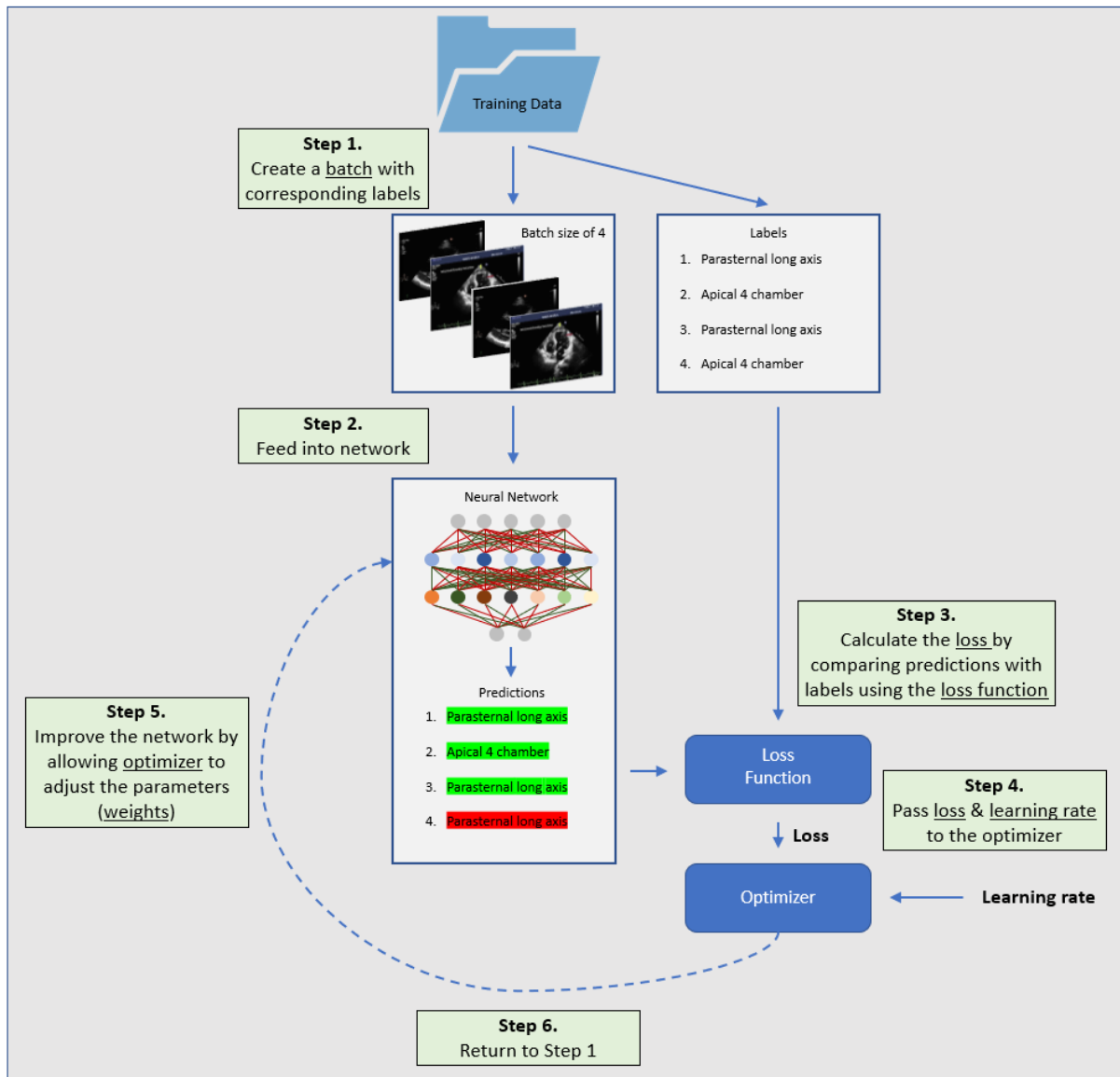


Figure 2-9 - The steps involved in training a neural network. **Step 1:** We load 4 examples of training data and the correct answers, as a batch. **Step 2:** The batch is fed into the neural network, which makes predictions for each of the 4 examples. **Step 3:** We calculate the loss by comparing the neural network's predictions with the labels, using the chosen loss function. **Step 4 & 5:** The optimizer uses the loss to adjust each of the weights in the neural network, with the size of the adjustment set by the user-supplied learning rate. **Step 6:** The cycle returns to Step 1, repeating until we have finished training.

In this process, a neural network is repeatedly shown a **'batch'** of **training data** from which it makes a prediction for each of them. These predictions are then compared with the corresponding correct answers (labels), using the **'loss function'**. This loss function quantifies how incorrect the neural network's predictions are. In the case of a classification task, to achieve 0 loss, the neural network would have to predict the correct class with 100% confidence, and 0% confidence in the other classes. Conversely, if it were to predict the wrong class with 100% confidence, then, with the loss function that we typically use for a classification task, the loss would be infinity.

Using calculus, we can calculate how the loss would change if we were to make a very small adjustment to each of the weights in the neural network. We supply these values for each weight, termed gradients, to the **optimizer** which then adjusts each of the weights in the neural network so that it can improve in performance. The size of the adjustment depends on the **learning rate** chosen.

After the neural network has seen the entire training dataset once (termed an **epoch**), we then assess its performance on the **validation dataset**, to identify its performance on data on which it has not been trained.

Next, the training set is reshuffled, and separated into a new combination of batches. This is important, so that none of these batches will be the same as before. Now the network goes through these batches, in the same way as before. This process continues until the neural network has seen the entire training dataset a prespecified number of times (the number of epochs).

With each epoch, the loss on the training set should fall, because the network is becoming increasingly familiar with the training set. Hopefully, the some of what it is learning is also applicable to other images outside the training data, but this cannot be assumed. We will gauge this applicability on outside images, using the validation dataset.

Once the validation loss stops improving, we consider the network trained as well as possible given the current settings. At this stage, a human researcher will evaluate the validation loss, and may wish to start all over again with a change in the design of the network, with the hope of achieving a lower validation loss. Changes that might be made include the **architecture** of the neural network, the **learning rate**, the **loss function**, or the **optimizer**.

Unfortunately, this process of the human researcher trying again does leave the researcher open to unintentionally picking an experiment which happened, by chance alone, to outperform the others. This is particularly so if the researcher tries very many times, or if they happen to have only a small validation dataset (whose results will therefore vary a lot even when only a few images are interpreted differently).

Therefore, finally, when the neural network's design and training strategy is finalised and the network has been fully trained, we assess its performance on a completely unseen **testing dataset**, and this is reported in the publication as the 'true' accuracy of the network.

2.9 Other strategies to improve convolutional neural network performance – reducing overfitting

When we train our neural network on the training dataset, we hope it is improving its performance by learning generalisable principles, rather than by simply memorising the particular images in the training set. The way we monitor this is by ensuring the performance on the validation dataset is improving during the training process. The accuracy on the training set will generally be better than that on the validation set, because of the inevitable tendency to do better on examples that the network has seen before. This excess score on the training set represents

“overfitting”. We try to avoid excessive overfitting, as it is not useful for the intended purpose of the network, which is to correctly interpret unseen data.

There are several strategies to minimise this tendency to overfit, the two main ones being data augmentation and regularisation.

Data augmentation is a method of training the network with more examples than we really have, by transforming each of the genuine training images in different ways so that it produces multiple images that are numerically different pixel content, and so it would be harder for the neural network to memorise them all.⁴⁸ This is done by randomly pre-processing each image on every pass through the training dataset. These random transformations include flipping, rotating, cropping, brightening, stretching and skewing the images. By combining these operations randomly on each image, a single original training image can look remarkably different to both the human eye and the neural network. Because this process effectively provides a larger more varied dataset, it is called data augmentation. An example of data augmentation is shown in Figure 2-10. In this figure, a single pacemaker image is randomly augmented to provide 16 further synthetic images from which the network can be trained.

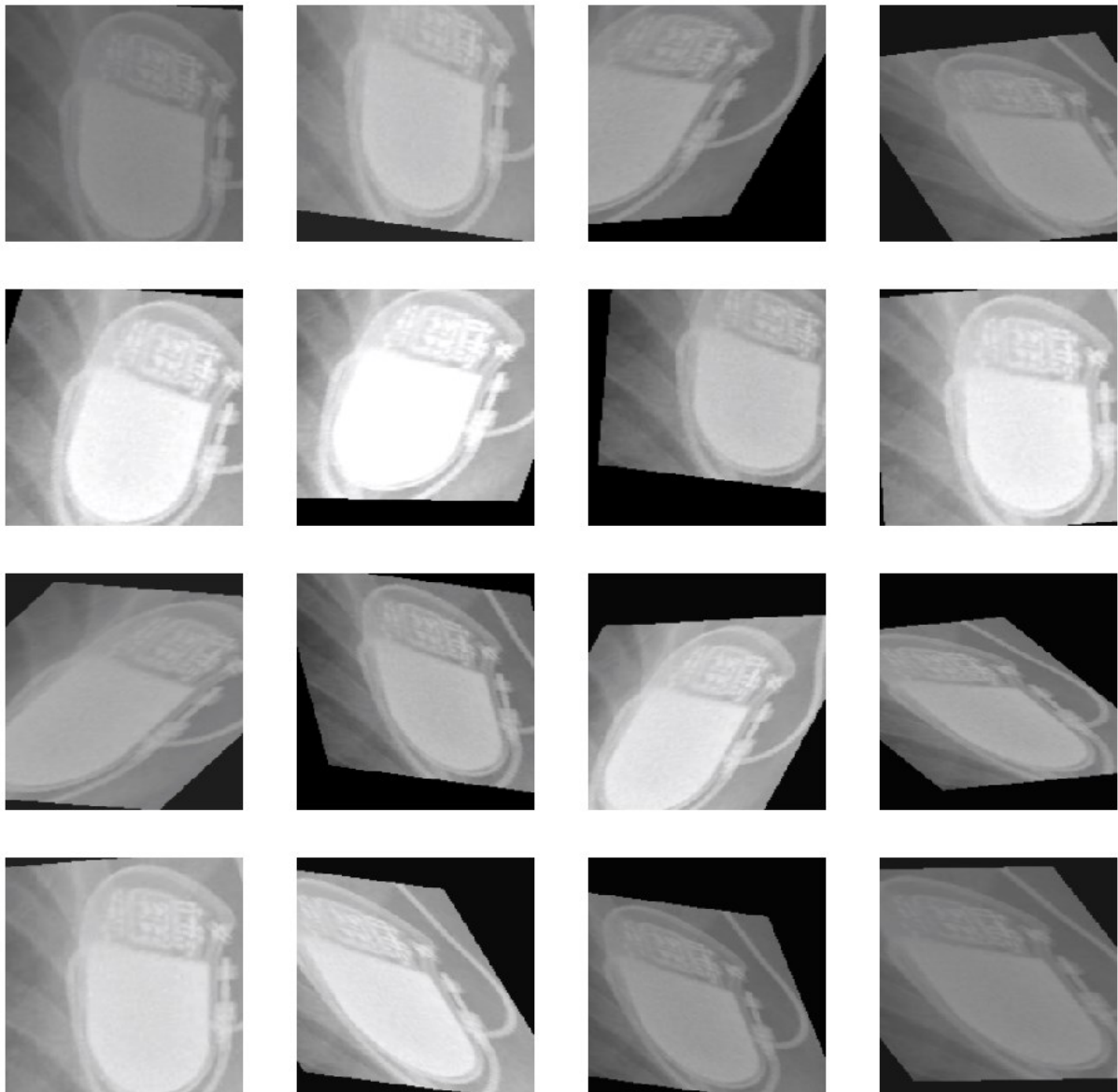


Figure 2-10 - A single picture of a pacemaker on a chest X-ray which has been randomly augmented. 16 different examples are shown, with random amounts of cropping, rotation, skewing and brightening. The black border reflects areas which lie outside of the original image.

Regularisation is another way to prevent overfitting. Here, instead of transforming the input images, adjustments are made to how the neural network learns. Each of these adjustments makes it harder for the neural network to simply memorise specific example images.

One way of doing this is to discourage the network from developing very large values at any particular position in the matrix of weights. A common way to do this is termed “L2 regularisation”, also known in the setting of CNNs as “weight decay”.⁴⁹ L2 regularisation assigns a penalty to parameters of the network which is proportional to the sum of the square to their values; this means that neural networks are favoured to rely on more parameters of moderate magnitude, rather than a few parameters very strongly. This makes it harder for the networks to overfit by learning a few very specific features within an image, as the regularisation encourages the network to rely on more features.

A bizarre form of regularisation is “dropout”.⁵⁰ Against all intuition, it is helpful during the training phase to randomly ignore parts of the network. This appears to make the network more robust as it can no longer depend on any particular part as a bottleneck in the decision process. This means it has to develop multiple ways of carrying out each task. It is akin to a sports team deliberately training with random players removed from the field, so that each of the skills needed for the team eventually develops in more than one player. In a neural network, this dropout is carried out by disregarding a certain proportion of the neurons on each training pass, as shown in Figure 2-11. This proportion is non-trivial, for example 40%. A dropped neuron provides no outputs to the next layer, and therefore is unable to contribute towards the final decision. Whilst in the short-term this is destructive, in the long-term it prevents the neural network from relying on a small number of neurons to govern the final decision.

A more subtle benefit of dropout is that it prevents small populations of neurons to “co-adapt”, i.e. become specifically depending on each other, which can worsen the tendency to overfit.

Using my analogy of the sports team, dropout helps prevent this because when team member A

is on the field, he or she needs to be able to work with any combination of other team members, and not get into a situation where they are hopelessly bereft when their long term teammate is not alongside them.

Dropout is only active during training time, because its purpose is to make the training more challenging, and therefore more useful. It is not used when we are validating or testing the network, or during the ultimate application. Again, this is analogous to training of the sports team: one might ask the team members to practice with weights on their legs or at altitude where oxygen levels are low, which reduces their performance during training, but makes them all the better when it comes to apply their skills practically.

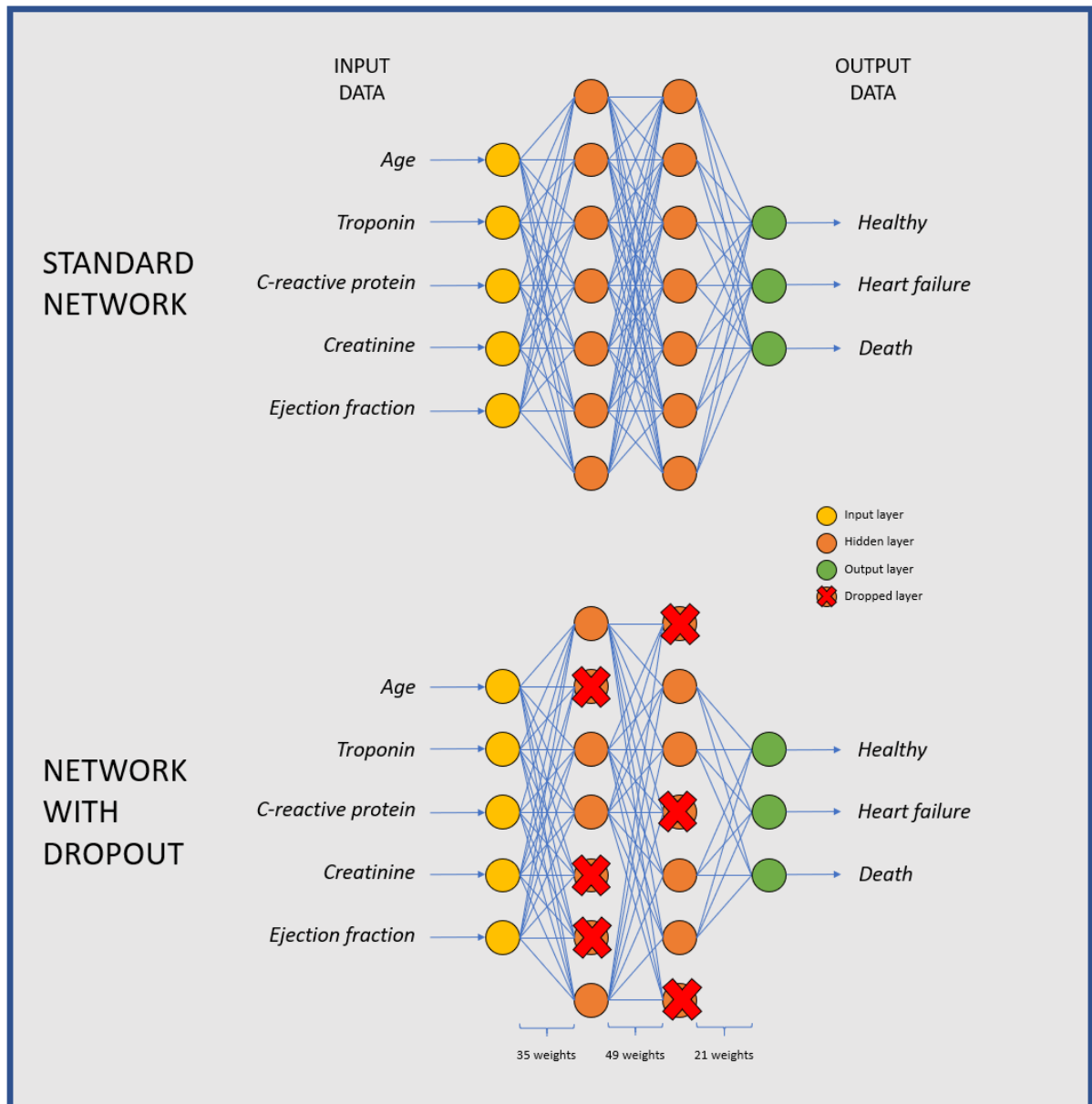


Figure 2-11 - A schematic showing two otherwise identical neural networks, one of which uses dropout. During training, there is a certain percentage (e.g. 40%) that any single neuron in the hidden layers will be “dropped”, and not contribute towards the neural network’s outputs. The neurons dropped changes with every batch. This punishing the neural network for relying on a small population of neurons to make the final decision, as there is a 40% chance of these not being present during training time. During testing time, however, dropout is disabled, so all neurons can be used to come to a decision. Dropout can be applied to any layer of the neural network, though in practice one would not normally apply it to the input or output neurons as the former would reduce 40% of the input data, and the latter would prevent 40% of the categories being predictable by the network.

2.10 Visualisation of learning - saliency mapping

One common criticism of neural networks, in comparison to other methods of computerised decision making, is that they are “black boxes”.^{51,52} This is because when a human feeds a sample of data into a neural network and gets a result, it is very difficult to usefully explain *how* the neural network came to that decision. It can always be explained mathematically, but this would require a formula with millions of symbols for all the weights, and would therefore not be useful for humans, who usually explain decisions on a handful of features at most. Because of this, several strategies have been employed to help understand how neural networks translate inputs (such as images) into decisions.

One approach is to take an image and see which parts, if you changed them, would cause the network to change its mind most dramatically. It is these parts of the input image that we consider ‘salient’. In practice, because the neural network operates algebraically, it is not actually necessary to create the modified images. Instead, the network of weights can be mathematically traced back from the decision to the original image. This approach, “saliency mapping”, was pioneered by Zisserman *et al.* in 2014.⁵³ This uses differential calculus to quantify how much each pixel in the input image contributes towards the final decision. One can then superimpose a map of these activities onto the original image, to allow the relevant parts of an image to be highlighted (Figure 2-12).

There are several reasons why saliency mapping may prove useful during neural network development.

The first reason is that saliency mapping may provide information about “how” the neural network is coming to its decision, which might further human understanding of a problem. This might be particularly illuminating during tasks where a neural network seems to achieve supra-human performance, and information could be gleaned which could inform future human

decision making. Indeed, I show an example of this in Section 4.4.5 and Figure 4-7 where saliency mapping highlights a specific circuit board component which allows a human to easily discern between two otherwise identical-looking pacemaker devices.

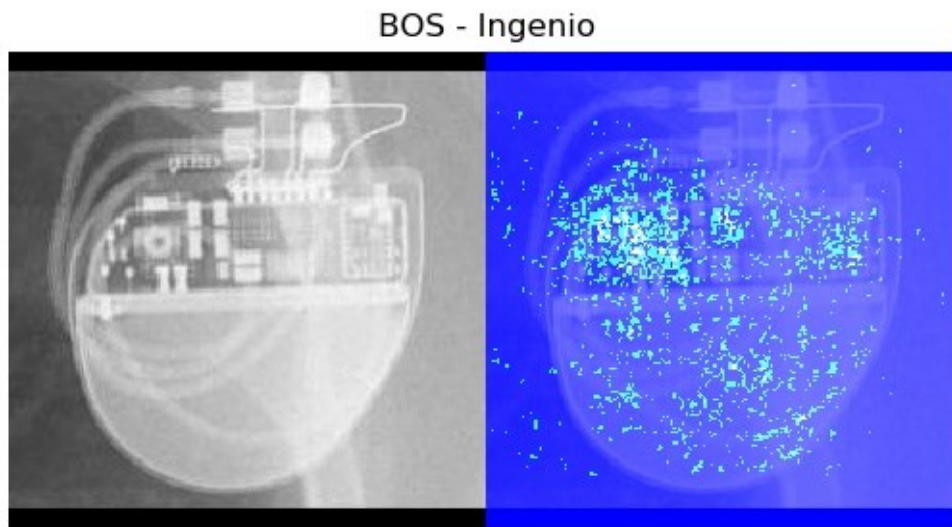


Figure 2-12 - A saliency map of a Boston Ingenio pacemaker present on a chest X-ray. The left side of the image shows the original chest X-ray. The right side of the image shows the saliency map overlying the original image. Royal blue regions represent pixels which are contributing little towards the neural network's decision, whilst brighter regions represent pixels which are greatly influencing the neural network's decision that this is a Boston Ingenio device.

A second important use of saliency mapping is to make sure that the network is making its decision from a sensible part of the image. Famously, there was a neural network that was trained to classify a picture as either containing a wolf or a huskie.⁵⁴ The accuracy of this neural network was excellent, with a performance greatly exceeding that expected by the team who trained it. However, when the team examined the network's decision-making using saliency mapping, it was not the characteristic curly tails of a husky that the network was focusing on - instead it was the presence or absence of snow! Images containing snow were invariably classified as containing a wolf, and those without snow were classified as huskies. This strategy led to impressive training set accuracy (and indeed, may even have translated into good 'real world' accuracy), but rendered it effectively useless if it was deployed to a webcam, say, overlooking a snowy field. Saliency mapping revealed what the network was relying on.

In general, saliency mapping helps ensure that a neural network's decision making is robust. To take an example, in Figure 2-12 we can see that almost all 'influential' pixels falls within the pacemaker device. Whilst this seems obvious, a common problem for neural networks is that they 'overfit' to training data, meaning they perform far better on images they were trained on (have seen before) than new images (Section 2.9). In the setting of convolutional neural networks, this can mean the neural networks have learned to recognise images by features which do not 'generalise' well to new examples. If Figure 2-12 had shown the neural network was using the patient's ribs or lung fields to come to its decision, this would be greatly concerning.

When saliency mapping identifies that the network is using an undesirable part of the image for its decision making, the researchers then have to acquire additional training images to prevent this from being the best solution for the training set. An obvious solution to the above example would be for the authors to add images to the training set of huskies in a snowy setting.

However, if all such examples were sled-pulling dogs, there is a risk the network will learn that "dogs in the snow are wolves rather than huskies, unless they have a collar or are attached to

reins”. Conversely, sourcing pictures of wolves within an urban setting is likely to be just as difficult.

Such examples illustrate why saliency mapping can prove useful during neural network training, but also why training datasets are typically very large: the hope is that in a large enough dataset, the neural network will learn that these confounding surrogate markers (such as wolves tending to be in the snow) are less accurate overall than more subtle cues of the dogs’ morphology.

2.11 The future of deep learning – optimism and scepticism

Deep learning will undoubtedly play an increasing role in cardiology and the recent explosion in the use can be attributed to three factors. First, the computing power required to train deep neural networks has only become available within the last decade. Second, the discovery of the effectiveness of more advanced networks such as CNNs is relatively recent; only in 2012 did a CNN, called AlexNet, become state of the art in the field of image classification.⁵⁵ Finally, training deep learning models typically requires large correctly-labelled datasets. In recent years there has been increasing focus on the generation of high-quality open curated datasets, such as the UK BioBank project.⁵⁶

However, concerns have been raised about the claims of some deep learning applications.

Overfitting, covered in section 2.9, is one of these concerns. This phenomenon, where a model such as a neural network performs much better on data it has seen before and been trained on than “real world” data, is common to many statistical methods, but neural networks are particularly prone to overfitting because they have millions of times more weights than conventional statistical models. These weights, in some sense, are learning to “memorise” data the network has seen before, and a neural network with millions of weights has a great capacity to do this. The hope is that in this memorization process that the neural network will learn general rules, such as “a car usually has 4 wheels”, that extend well to cars in the real world.

However, if the training data for a neural network learning to classify vehicles comprises only black cars with fixed roofs, it will likely perform very badly when faced with a red convertible. This will manifest as the neural network having a high accuracy during training, but much poorer when employed practically. Overfitting can be minimized by using very large datasets and by assessing the neural network on a random selection of the data against which it has not been trained, termed the “test set”. However, these test sets typically arise from the same source as the training dataset. It would be common practice for researchers to acquire, say, 8000 images and

split them at random into 6000 for training and validation, and 2000 for testing. If the original 8000 images had biases, such as a lack of convertibles, or huskies on snow, then the testing set would be just as biased and the resulting network would unexpectedly fail when it was shown a husky in a snowy field.

A second criticism often faced by deep learning is that neural networks do not enhance human understanding.^{51,52} However, there is now an increasing focus on opening up the black boxes of neural networks. Saliency mapping, for example, allows scientists to visualize which aspects of an image are most contributing to a decision.⁵³ I use this method in section 4.4.5 of this thesis.

Similarly, Google’s DeepDream works to adjust an image so that it better maximizes the activity of a chosen layer of neurons within a CNN.⁵⁷ This allows the roles of the different layers of a CNN to be indirectly visualized, as the features they find ‘interesting’ become increasingly over-represented.

Finally, some have aired frustration at the relative paucity of deep learning projects moving from “bench to bedside”.⁵⁸ Despite the large number of publications in the field, few deep learning models have entered clinical practice. The U.S. Food & Drug Administration have recently sought input from stakeholders on this issue, however, to help streamline the approval of AI in “Software as a Medical Device”.⁵⁹

For these reasons, I am passionate that all the systems I develop in this thesis are free for public use, and so all of the neural networks have been made freely accessible online via my personal website at www.james.dev.

3 Methods – Software & Hardware

3.1 Software

3.1.1 Machine learning programming languages and libraries

The algorithms used in deep learning could theoretically be applied using any modern programming language. However, due to the scale of the calculations necessary, these algorithms have to be written in a highly efficient and specific manner which allows them to be run on graphical processing units (GPUs; section 3.2). A small number of languages have had large quantities of optimised code developed, termed “libraries” or “frameworks”, which make deep learning using GPUs straightforward.

The Python programming language (The Python Software Foundation)⁶⁰ is the most commonly used language in the field of deep learning.⁶¹ Python is well suited towards research programming because it is an interpreted language, as opposed to a compiled language. Interpreted languages can be run immediately after code is written or altered, without needing to wait for the entire project to be converted into machine-readable code before the program starts (a process termed ‘compiling’). Python’s ability to run without the often-time-consuming compilation step facilitates rapid development. The downside of interpreted languages is that they are typically slower *during* running, because the code is, in effect, compiled as the user runs it. However, python has several deep learning libraries/frameworks specifically created for developing neural networks, including Tensorflow,⁶² Keras⁶³ and Pytorch.⁶⁴ These are written in pre-compiled code which the user can use as building blocks to develop their neural networks, providing the speed of a compiled language and the ease of use of an interpreted language.

In this thesis I use the Python programming language (version 3) to develop the neural networks and I use a mixture of Keras/Tensorflow (Chapter 4 and 6) and Pytorch (Chapters 5 and 7) across the four experimental chapters.

3.1.2 Statistical programming

Throughout the four experimental chapters I employ a variety of statistical methods. These are covered in detail in each chapter's dedicated methods section, but typically utilise the programming language "R".⁶⁵ R, like Python, is an interpreted programming language, but has an emphasis on statistical computing. It also has a number of powerful graphing libraries, such as ggplot2,⁶⁶ which has been used to develop many of the figures contained in this thesis.

3.1.3 Integrated development environments

Programming in Python can be performed with any text editor on a modern computer. However, in large projects with multiple separate code files, an integrated development environment (IDE) can increase productivity by helping organise files, identifying common errors, and providing 'debug' tools which assist in tracking down software bugs. Throughout my thesis I have used the PyCharm IDE (Figure 3-1; JetBrains, Prague, Czech Republic) for Python development.

For R development, I have used the RStudio IDE (Figure 3-2; RStudio Inc., Boston, Massachusetts, USA).

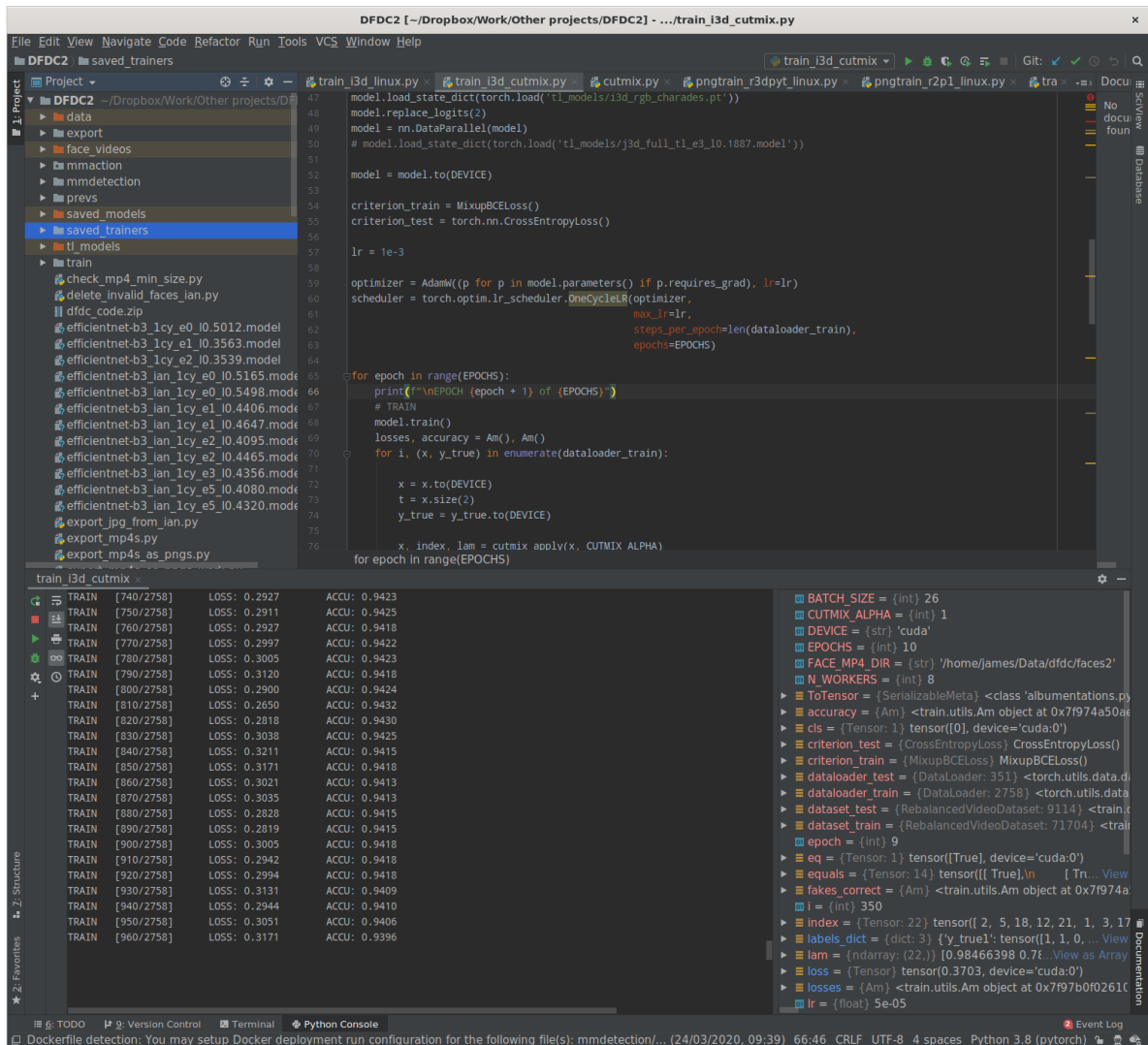


Figure 3-1 - An example of the PyCharm integrated development environment (IDE). The top left panel of the screen shows files that are part of the current project. The top right of the screen shows the Python code contained in a selected file. The bottom left shows the status of a program which is currently running. The bottom right shows the data currently being used by that program.

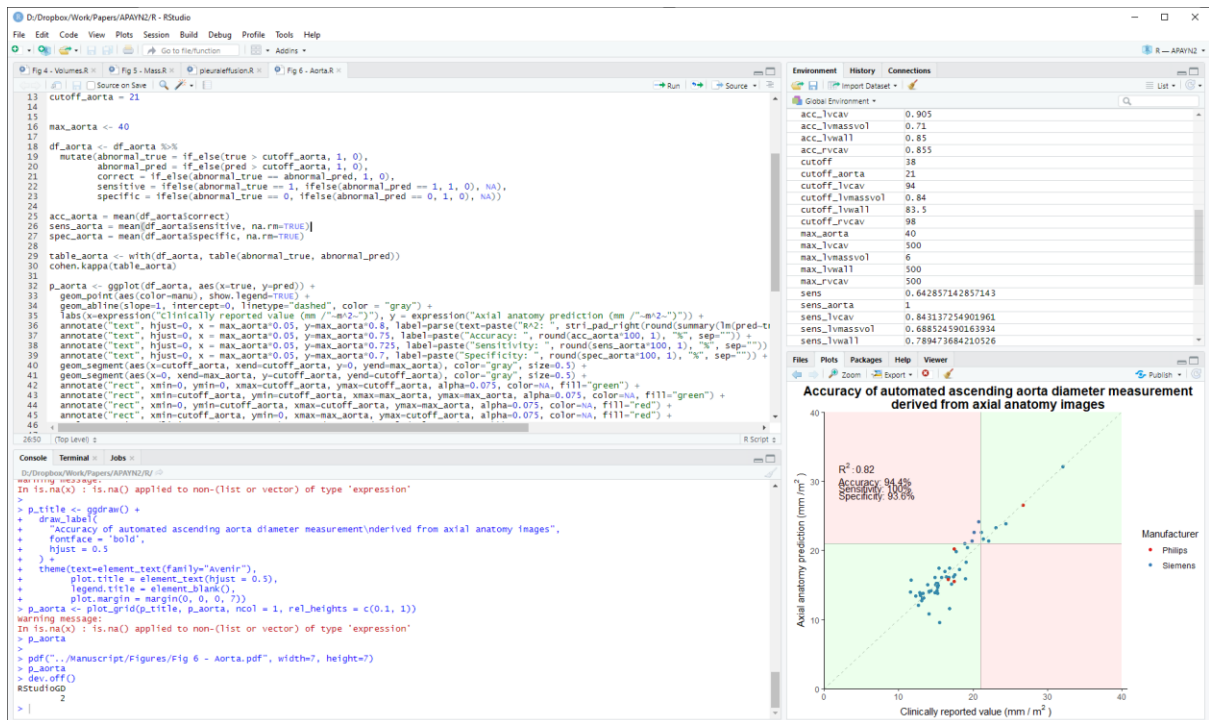


Figure 3-2 - The RStudio integrated development environment (IDE). The organisation is similar to that of PyCharm, with source code visible on the top left of the screen. The bottom left shows the 'output' of the program as it runs. The top right panel shows the current variables in the computer's memory that contain data. The bottom right demonstrated a plot that I have created using the visible code, and which forms part of section 7.4.3.

3.2 Deep learning hardware (graphical processing units)

Graphical processing units (GPUs) are specialised electronic circuits, which typically take the form of additional circuit boards which can be added to a desktop computer (Figure 3-3). As their name implies, they were developed to greatly improve the image processing capabilities of computers, allowing 3-dimensional video games to be developed. However, the mathematical operations involved in computer graphics that GPUs excel at are the same as those used in deep learning: matrix mathematics. Modern deep learning workstations typically comprise one or more GPUs, and training data is repeatedly exchanged between the computer's main central processing unit (CPU) and each GPU during neural network training. During my thesis I initially used two Nvidia 1080 Ti GPUs, and later two Nvidia RTX Titan GPUs (Figure 3-4).



Figure 3-3 - Example of an Nvidia graphic processing unit (GPU) - the RTX Titan model is shown. Image Courtesy of Nvidia Corporation, California, United States.



Figure 3-4 - The workstation used by the Student. Two RTX Titan graphical processing units (GPUs) are visible below the large central processing unit (CPU) cooler. The power supply at the bottom of the case shows the current power draw for the system in Watts.

4 Identification of cardiac implantable electronic devices from chest X-rays

The research covered in this chapter has been previously published as:

Howard JP, Fisher L, Shun-Shin MJ *et al.*

“Cardiac Rhythm Device Identification Using Neural Networks.”

JACC Clin Electrophysiol. 2019;5(5):576–586. doi:10.1016/j.jacep.2019.02.003.⁶⁷

The work was also presented at the Imperial College Clinical Academic Training Conference 2019, where it was awarded the prize for Best Presentation.

4.1 Abstract

4.1.1 Background

Cardiac implantable electronic devices (CIEDs) include pacemakers, defibrillators and implantable loop recorders, and over a million of these devices are inserted every year. When a patient with a CIED presents to hospital as an emergency, medical staff may need to urgently interrogate or reprogram the device. However, this can only be performed if the medical staff know the manufacturer of the CIED, as these processes require that the correct proprietary equipment to be brought to the patient's bedside.

Flow charts have been developed to allow physicians and physiologists to identify the manufacturer of a CIED from its X-ray appearance. However, this process can be painstaking, and the flow-charts become obsolete if they are not updated as new CIEDs are developed.

I therefore set out to develop an automated system to identify what pacemaker is present in an X-ray.

4.1.2 Methods

I extracted 1676 X-ray images of CIEDs from Hammersmith Hospital. These comprised 45 different models from 5 manufacturers. I developed a convolutional neural network to classify these images and trained it using 1451 of the images (the training set). The performance of the neural network was assessed on a smaller testing set of 225 images, comprising 5 examples of each model. The performance of the neural network was compared with that of 5 cardiologists using a previously published flow-chart.

4.1.3 Results

The convolutional neural network was 99.6% (95% CI 97.5 to 100) accurate in identifying the manufacturer of a device from an X-ray, and 96.4% (95% CI 93.1 to 98.5) accurate in identifying the model group. The median accuracy of the cardiologists using the human flow-chart was 72% (range 62.2% to 88.9%). Identification of the model group was not possible using the flow chart. The convolutional neural network was superior to each of the cardiologists in identifying the manufacturer ($p < 0.0001$ against the median human; $p < 0.0001$ against the best human).

4.1.4 Conclusions

A convolutional neural network is able to identify the manufacturer of a CIED present on a chest X-ray more accurately than a cardiologist using a flow-chart. Furthermore, the network can accurately identify (with 96.4% accuracy) the exact model group of the device. This system may facilitate the interrogation and reprogramming of CIEDs in the emergency setting and therefore speed up diagnosis and treatment of patients with these devices. The neural network is publicly available and free to use at <http://ppm.jph.am>.

4.2 Introduction

4.2.1 Interrogation and reprogramming of cardiac implantable electronic devices

Cardiac implantable electronic devices (CIEDs) include pacemakers, defibrillators and implantable loop recorders. The numbers of CIEDs implanted continues to rise,³ with over 1 million devices inserted internationally every year.⁴

It is therefore increasingly common for a critically unwell patient presenting to hospital to have been implanted with a CIED. Specialised staff can use computer equipment to “interrogate” these devices and gain valuable electrophysiological information, such as the presence of any arrhythmia. Furthermore, CIEDs such as implantable cardioverter defibrillators (ICDs) can be “reprogrammed” to either deliver lifesaving therapies (such as anti-tachycardia pacing or a defibrillator shock) or withhold them in the settings of a device malfunction.

However, for medical staff to interrogate and reprogram a CIED, they must know the device’s manufacturer so the specific proprietary computer can be brought to the patient’s bedside. If the medical team do not have access to the implantation records, or the patient cannot provide the medical staff with this information, pacemaker interrogation and reprogramming may be delayed. In patients presenting with haemodynamically unstable arrhythmia, or inappropriate defibrillator therapies, these delays may be harmful.

4.2.2 Current approaches for identification of CIEDs

Up to 80% of physicians have reported having difficulties with identifying CIEDs.⁵ For this reason, efforts have been made to assist the medical team in identifying devices when medical records are not available. A logical approach is to look at a device’s unique appearance on a chest

X-ray. Unfortunately, two different CIED models from a single manufacturer can look dramatically different on a chest X-ray (Figure 1-3). For this reason, a physician or cardiac physiologist must not merely be able to identify one X-ray appearance per manufacturer, but many more.

Because of these difficulties, manual flow charts have been developed. However, these flow charts are limited in effectiveness for three reasons. First, they comprise multiple steps, making them more time consuming and difficult to use than an automated system (Figure 1-4). Second, the most recent flow chart is 8 years old,⁵ and so any devices released since then with novel X-ray appearances are not classifiable using it. Third, even at the time of publication, the study authors reported accuracies of only 90%.⁵

I therefore wished to create an automated system which could process a chest X-ray of a CIED and accurately identify the manufacturer of the device. I also wished to explore whether such a system could identify the exact *model* of a CIED. Such a system could be useful for identifying whether a device is compatible with an MRI scanner, for example.

4.3 Methods

4.3.1 Data extraction

I extracted pacemaker implantation records from Imperial College Healthcare NHS Trust’s pacing database, for procedures between February 1998 and May 2018. Eligible CIEDs for inclusion were those with at least 25 X-rays from unique patients. I included both portable and departmental AP/PA chest X-rays. Lateral chest X-rays were not included. In the absence of any data outlining the prevalence of different devices across the world, I generated a dataset where all types of device were represented in roughly equal proportions; images were therefore extracted from consecutive patients to a maximum of 40 images per model to minimize class imbalance.⁶⁸

I noted during extraction that in several cases, when a manufacturer introduced a new model, there was no detectable change on the X-ray. This may represent a purely software change or replacement of parts by visually indistinguishable replacements. I therefore grouped models with identical appearances into ‘model groups’.

5 images from each of the CIED model groups were randomly allocated to the testing dataset. This dataset was not shown to the neural network at any stage of its training and was only used to report the network’s final accuracy.

The remaining 20 to 35 images of each model group were allocated to the training dataset. This dataset was used during two stages: network design, and final model training.

For the first stage (“network design” in Figure 4-1) I assess each candidate neural network design by its ability to learn on 75% of the training set and correctly make predictions on the remaining 25% of the training set (termed the validation set). This is done four times, so that all of the training set participates in turn in both roles. This process is termed ‘4-fold cross-validation’ (unrelated to the final testing which is performed on the completely separate ‘test set’).

The second stage (“final model training”) begins with the neural network design chosen by the first stage. This starts with a fresh neural network with no prior exposure to device images. The network is then trained on the entire training set, resulting in the final trained network.

Finally, this final network is exposed for the first time to the ‘test set’, which has been kept separate throughout. The network assessed for its ability to correctly classify manufacturer and model group on test set cases.

Regulatory and ethical approval for the study was granted by the Health Research Authority (Integrated Research Application System identifier 249461).

4.3.2 Data processing

From each chest X-ray image, I extracted a square region of interest slightly larger than the device. This maximized the signal-to-noise ratio for the network and guaranteed anonymization. These cropped images were then resized to 224 by 224 pixels and rescaled to yield pixel values between 0 and 1.

4.3.3 Convolutional neural network architecture and training

I assessed 5 different convolutional neural network architectures (DenseNet, Inception V3, VGGNet, ResNet and Xception), all of which have at some stage in recent years been the world-leading design on the ImageNet dataset challenge.^{69–73} All networks were initialized using weights derived from training on ImageNet, before the whole model was retrained. Regularization and Dropout were implemented for each network according to their original publications.

For each network, the output layer was set to have 45 output neurons (one for each device model group). Loss was calculated over batches of 16 images using the categorical cross entropy

loss function and weights were updated using the Adadelta optimizer.⁷⁴ Loss is a technical marker which is used to assess the network's performance and make improvements. Loss is more sensitive than simply the misclassification rate (the inverse of accuracy) because to achieve full marks (zero loss) the network has, for each pacemaker image, to be 100% confident in the correct prediction and 0% suspicion for all 44 other model groups. Training of the neural network is an automatic process of adjusting the synapse weights to minimize this loss. Training continued until validation loss plateaued (15 epochs). I augmented training with random rotation, width- and height-shifting, vertical and horizontal flipping, shearing and zooming. Programming was performed with the Python programming language, with the Tensorflow⁶² and Keras⁶³ machine learning frameworks. Training was performed on 2 Nvidia GeForce GTX 1080 Ti graphical processing units.

4.3.4 Visualisation of learning

Examples of each model group were processed to provide saliency maps⁵³ where the pixels with the highest gradient with regards to the correct class (i.e. the pixels contributing most to the decision of the network) are highlighted. This was performed using the keras-vis package.⁷⁵

4.3.5 Human expert performance using the flow chart

The test set of 225 images were supplied to 5 independent cardiologists (of which two were electrophysiologists) along with the full manuscript of the CaRDIA-X algorithm, the most recent algorithm for classifying cardiac devices from chest X-rays.⁵ The algorithm only aims to distinguish between manufacturers rather than identifying the particular model group. With each image, graders were informed whether the device was a pacemaker, defibrillator or loop recorder. Graders were asked to classify each device as Biotronik, Boston Scientific (including

Guidant and Cameron Health), Medtronic (including Vitatron), Sorin (including Liva Nova) or St Jude (now Abbott).

4.3.6 Statistical analysis

The prespecified primary endpoint was a superiority test for manufacturer accuracy between the artificial network and human expert graders using the CaRDIA-X manual algorithm. The testing was by McNemar's test with $p = 0.05$ as the threshold for statistical significance, with an exact test used for contingency tables including any counts below 25. Because each human grader is an individual, the primary endpoint was calculated with reference to the human grader with the median accuracy.

Accuracy was defined as the number of correctly-classified images in the test set divided by the total number of images in the test set. Confidence intervals for accuracy were calculated using the exact binomial method. For manufacturer accuracy, where class sizes were inherently unequal, we also calculated the F_1 score, defined as double the harmonic average of the precision and recall, bounded between 0 and 1.

I assessed the accuracy of the network across several subgroups (departmental versus portable X-rays, pacemakers versus ICDs, and across the different device manufacturers) using Fisher's exact test. Welch's unequal variances t-test was used to assess for differences in image quality (sharpness) between departmental and portable X-rays by calculating the variance of Laplacian for each image.⁷⁶

Statistical analysis was performed using the R programming language.⁶⁵

4.4 Results

4.4.1 Dataset

The full dataset comprised 1676 images of unique devices from 1575 unique patients (some patients had more than one device during the study period). Although there were 66 different nominal device models, several of these were visually indistinguishable from each other, perhaps representing purely software changes between models. There were 45 different model groups with unique X-ray appearances (Table 4-1). 278 (16.6%) of the X-rays were from portable X-ray machines; the remaining 1398 (83.4%) were departmental AP or PA X-rays.

The test set was made up of 5 examples of each of the 45 final model groups to make a total of 225 examples. 38 (16.9%) of these were portable radiographs. The remaining 1451 cases were assigned to the training set.

The study flow chart is shown in Figure 4-1.

Manufacturer	Nominal model	n	Model group	n
Pacemaker (n = 1213)				
Biotronik	Actros	20	Actros / Philos	40
	Philos	20		
	Cyclos	27	Cyclos	27
	Evia	28	Evia	28
Boston Scientific	Altrua	20	Altrua / Insignia	40
	Insignia	20		
	Contak Renewal TR2	40	Contak Renewal TR2	40
	Contak TR	10	Contak TR / Discovery / Meridian / Pulsar Max	40
	Discovery	10		
	Meridian	10		
	Pulsar Max	10		
	Ingenio	40	Ingenio	40
	Proponent	40	Proponent	40
	Visionist	40	Visionist	40
Medtronic	Adapta	10	Adapta / Kappa / Sensia / Versa	40
	Kappa	10		
	Sensia	10		
	Versa	10		
	Advise	40	Advise	40
	AT500	38	AT500	38
	Azure	40	Azure	40
	C20	20	C20 / T20	40
	T20	20		
	C60	40	C60	40
	Enrhythm	40	Enrhythm	40
	Insync III	40	Insync III	40
	Sigma	40	Sigma	40
	Syncra	40	Syncra	40
	Vita II	29	Vita II	29
Sorin	Elect	40	Elect	40
	Elect XS Plus	30	Elect XS Plus	30
	MiniSwing	28	MiniSwing	28
	Neway	37	Neway	37
	Reply	40	Reply	40
	Rhapsody	20	Rhapsody / Symphony	40
	Symphony	20		
	Thesis	36	Thesis	36
St Jude	Accent	40	Accent	40
	Allure Quadra	40	Allure Quadra	40
	Identity	40	Identity	40
	Victory	40	Victory	40
	Zephyr	40	Zephyr	40
ICD (n = 415)				
Boston Scientific	Autogen	10	Autogen / Cognis / Energen / Teligen	40
	Cognis	10		
	Energen	10		
	Teligen	10		

	Contak Renewal 4	40	Contak Renewal 4	40
	Emblem	40	Emblem	40
	Ventak Prizm	33	Ventak Prizm	40
	Vitality	40	Vitality	40
Medtronic	Claria	13		
	Evera	13	Claria / Evera / Viva	40
	Viva	14		
	Concerto	8		
	Consulta	8	Concerto / Consulta /	
	Maximo	8	Maximo / Protecta /	40
	Protecta	8	Secura	
	Secura	8		
	Maximo	30	Maximo	30
Sorin	Ovatio	25	Ovatio	25
St Jude	Ellipse	40	Ellipse	40
	Quadra Assura	40	Quadra Assura	40
Loop recorders (n = 58)				58
Medtronic	Reveal	26	Reveal	26
	Reveal Linq	32	Reveal Linq	32

Table 4-1 - Distribution of classes across the entire dataset, broken down by device type, manufacturer and model. Visually identical model names (middle column) are merged into 'model groups' (right column).

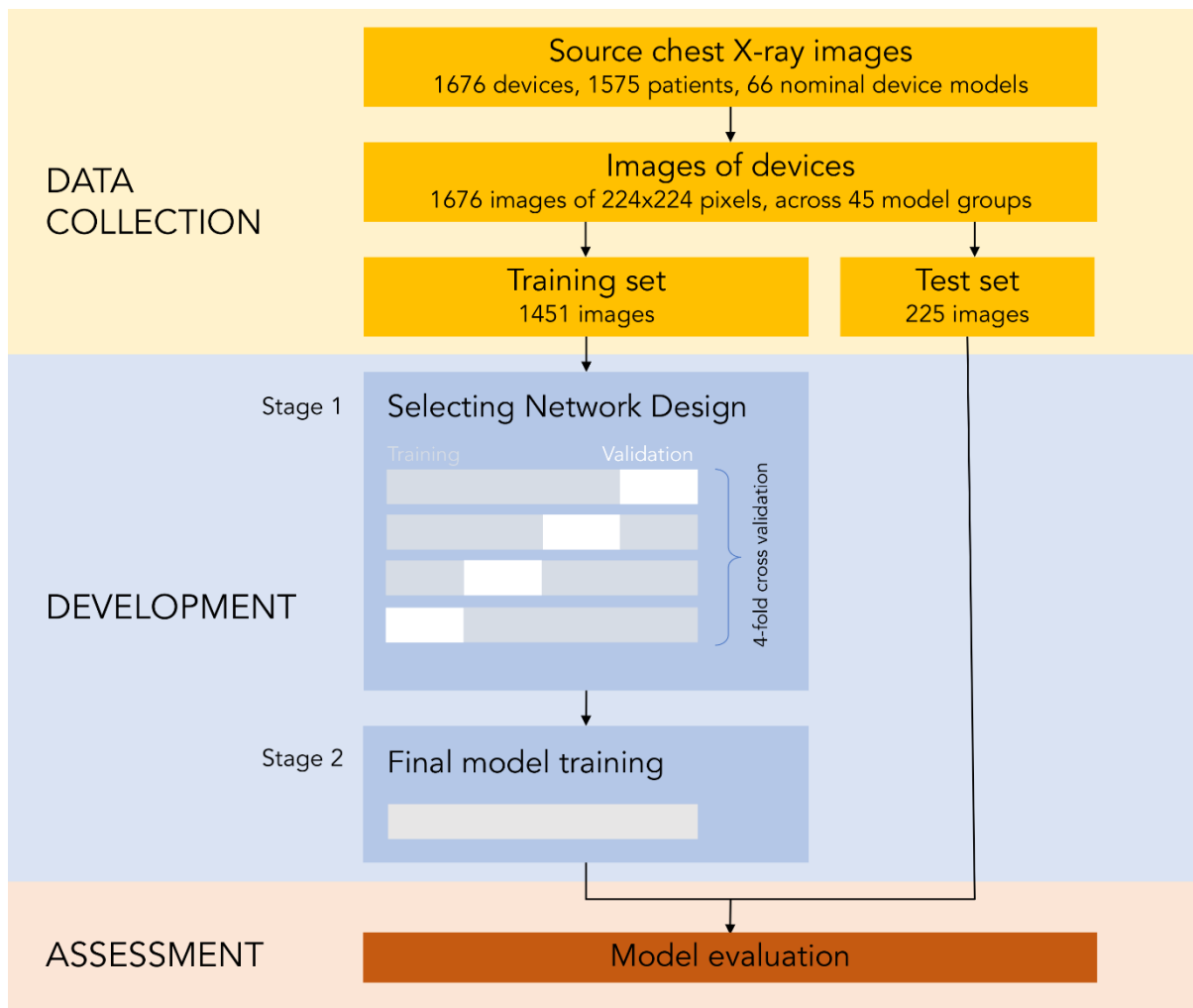


Figure 4-1 - Study design flowchart. The study was designed in three phases, comprising data collection, development of the neural network, and assessment of the network. The development of the neural network was broken down into two stages. Stage 1 involved selecting the optimal network design. Stage 2 involved training the 'final' model against the entire training set, and then assessing its performance on the unseen 'test set', allowing comparison versus humans.

4.4.2 Stage 1: Neural network architecture development

For all the network designs, after 15 epochs of training in Stage 1, the network had reached a plateau of performance, manifesting as a plateau after the initial decline in the validation loss. Table 4-2 shows the level of this plateau in the validation loss, and corresponding validation accuracy, for each of the five network architectures assessed. Each number displayed is the averaged value over the four splits of the training set. The accuracy varied from 4.4% for VGGNet to 91.1% for Xception.

Based on these results, the conclusion of Stage 1 was to select the Xception architecture for Stage 2, and to pre-specify that the number of epochs of training would be 15. Stage 2 therefore began with a fresh neural network of the Xception architecture and performed 15 epochs of training using the full training set of 1451 images.

Finally, the ‘test set’ of data which had been set aside until now was tested, once, using the final neural network produced by Stage 2.

Architecture	Trainable parameters	Loss (lower is better)	% Accuracy (higher is better)
DenseNet 121 ⁷²	7.0 million	0.36	90.8
Inception V3 ⁷⁰	21.9 million	1.06	79.5
Resnet50 ⁷¹	23.6 million	3.24	44.9
VGGNet 16 ⁶⁹	14.7 million	4.33	4.4
Xception ⁷³	20.9 million	0.34	91.1

Table 4-2 - Results of Stage 1, in which the 5 architectures are compared having been trained on only three quarters of the training data at a time. Performance of five network designs. Loss is a measure of inaccuracy which gives penalizes confident wrong predictions more than unconfident ones.

4.4.3 Stage 2: Final neural network training and assessment

The accuracy of the final neural network for identifying the manufacturer of a device was 99.6% (95% CI 97.5 to 100.0), corresponding to an F_1 score of 0.996. The performance is shown as a confusion matrix in Figure 4-2. The only image wrongly classified was of a Medtronic Adapta device which was mistaken for a Sorin Reply (Figure 4-4).

Inevitably, performance for identifying the model group (rather than only the manufacturer) was lower. Accuracy was 96.4% (95% CI 93.1 to 98.5) and the F_1 score was 0.964. Figure 3 shows the confusion matrix. The 8 images for which the neural network suggested an incorrect model group are all shown in Appendix 2, along with the top 3 predictions by the neural network for each example. Notably, in 7 out of 8 of these, the correct model group was one of the top 3 predictions. Therefore, what is commonly described as the ‘top 3’ accuracy was 99.6% (95% CI 97.5 to 100.0) for model group.

The model group accuracy for portable X-rays was 89.5% (95% CI 75.2 to 97.1) versus 97.9% (95% CI 94.6 to 99.4) for departmental X-rays ($p = 0.029$ for difference between the two groups). This corresponded with departmental X-ray images being significantly sharper as judged by the variance of Laplacian compared to portable X-rays ($p < 0.0001$ for difference between the two groups). The single manufacturer classification error was of a departmental X-ray, however. Model group accuracy for pacemakers was 95.0% (95% CI 90.4 to 97.8) versus 96.4% (95% CI 87.5 to 99.6) for ICDs ($p = 1.00$ for difference between the two groups). Model group accuracy did not vary significantly between different manufacturers ($p=0.954$).

Manufacturer-specific Neural Network Performance

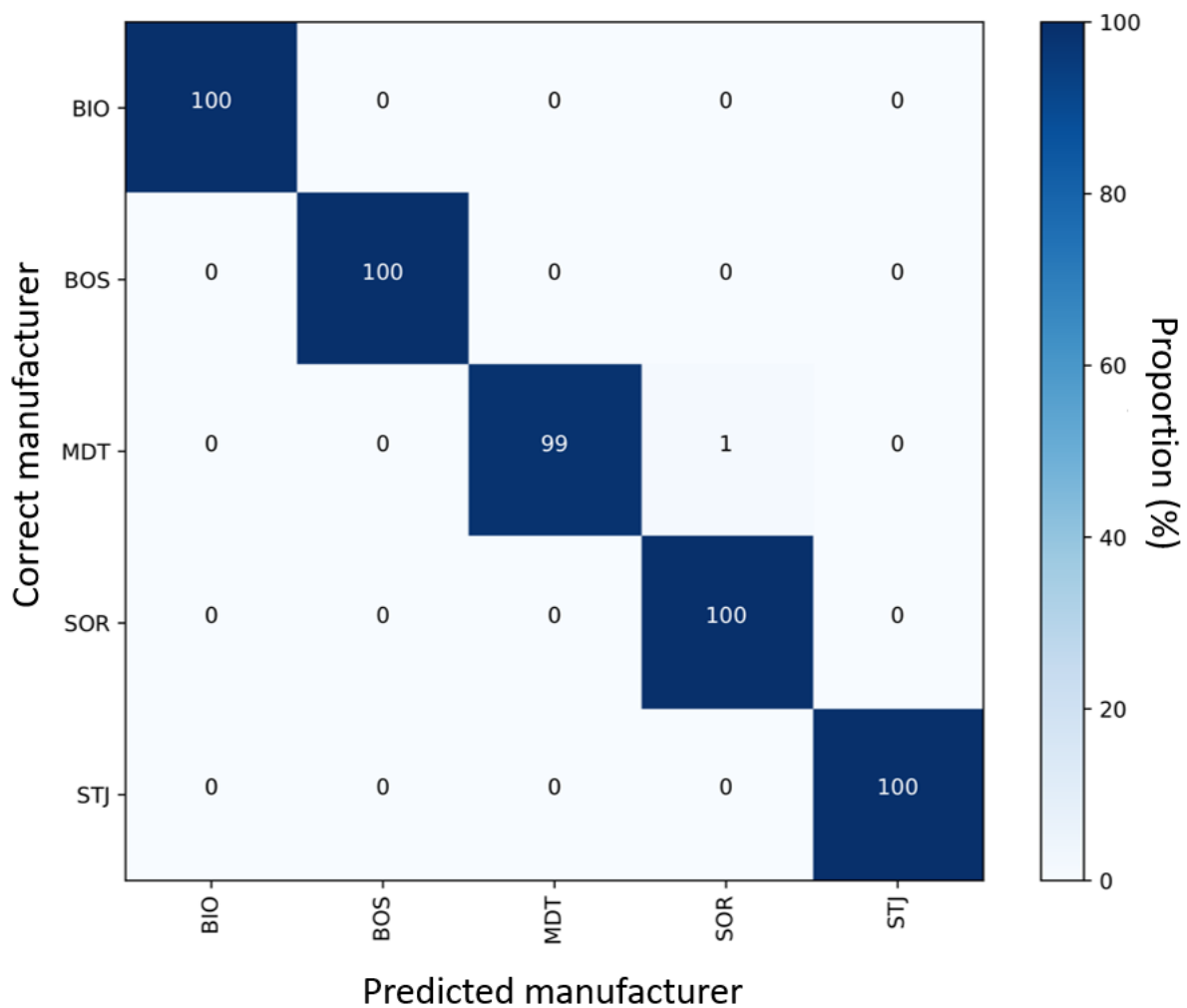


Figure 4-2 - Neural network performance. Confusion matrix showing the accuracy of the network in predicting the correct manufacturer of devices. BIO=Biotronik; BOS=Boston Scientific; MDT=Medtronic; SOR=Sorin; STJ=St. Jude.

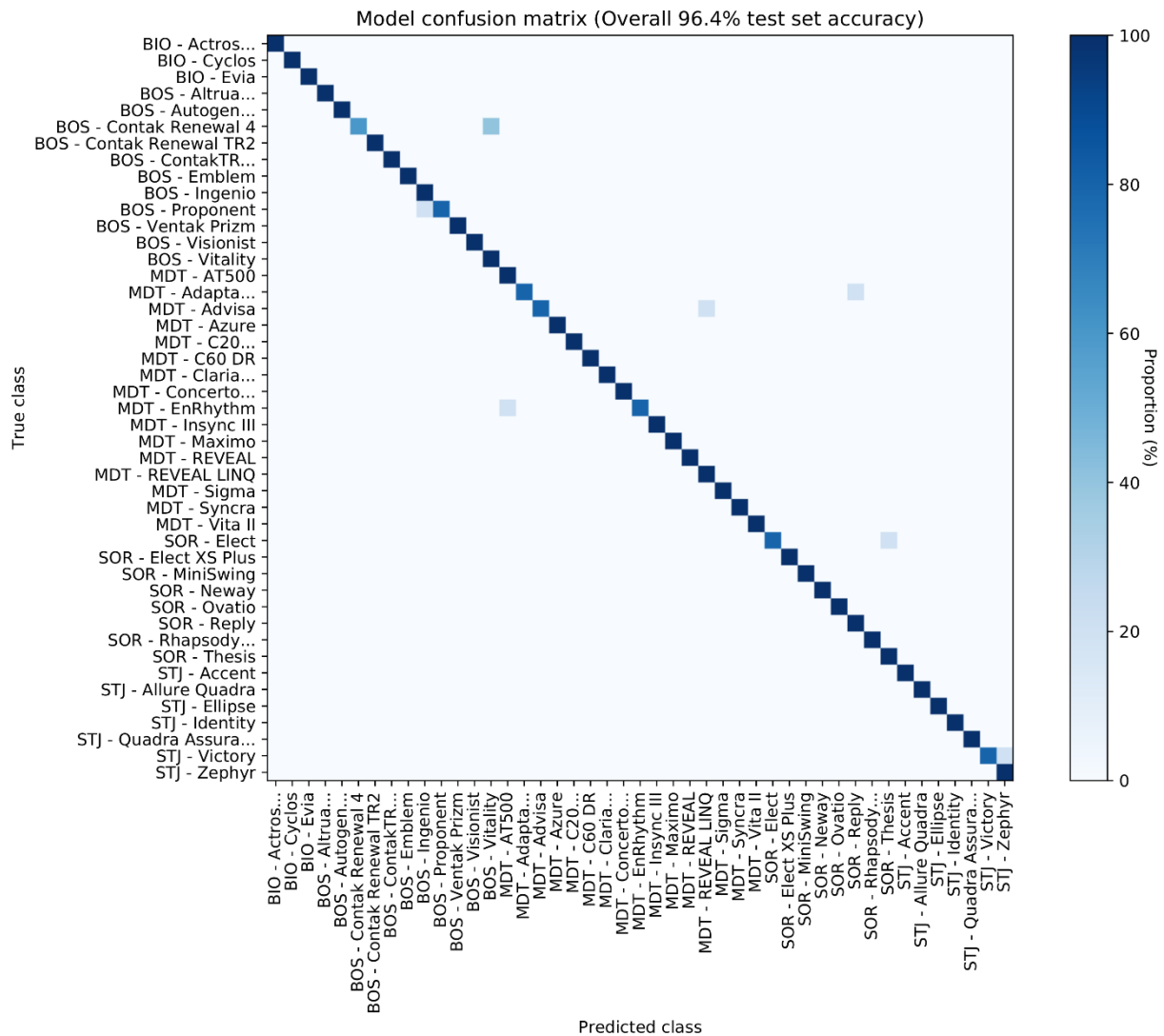


Figure 4-3 - Confusion matrix showing the accuracy of the network in predicting the correct model of devices. Class names ending in ‘...’ refer to those including more than 1 device type with identical appearance. BIO=Biotronik; BOS=Boston Scientific; MDT=Medtronic; SOR=Sorin; STJ=St. Jude.

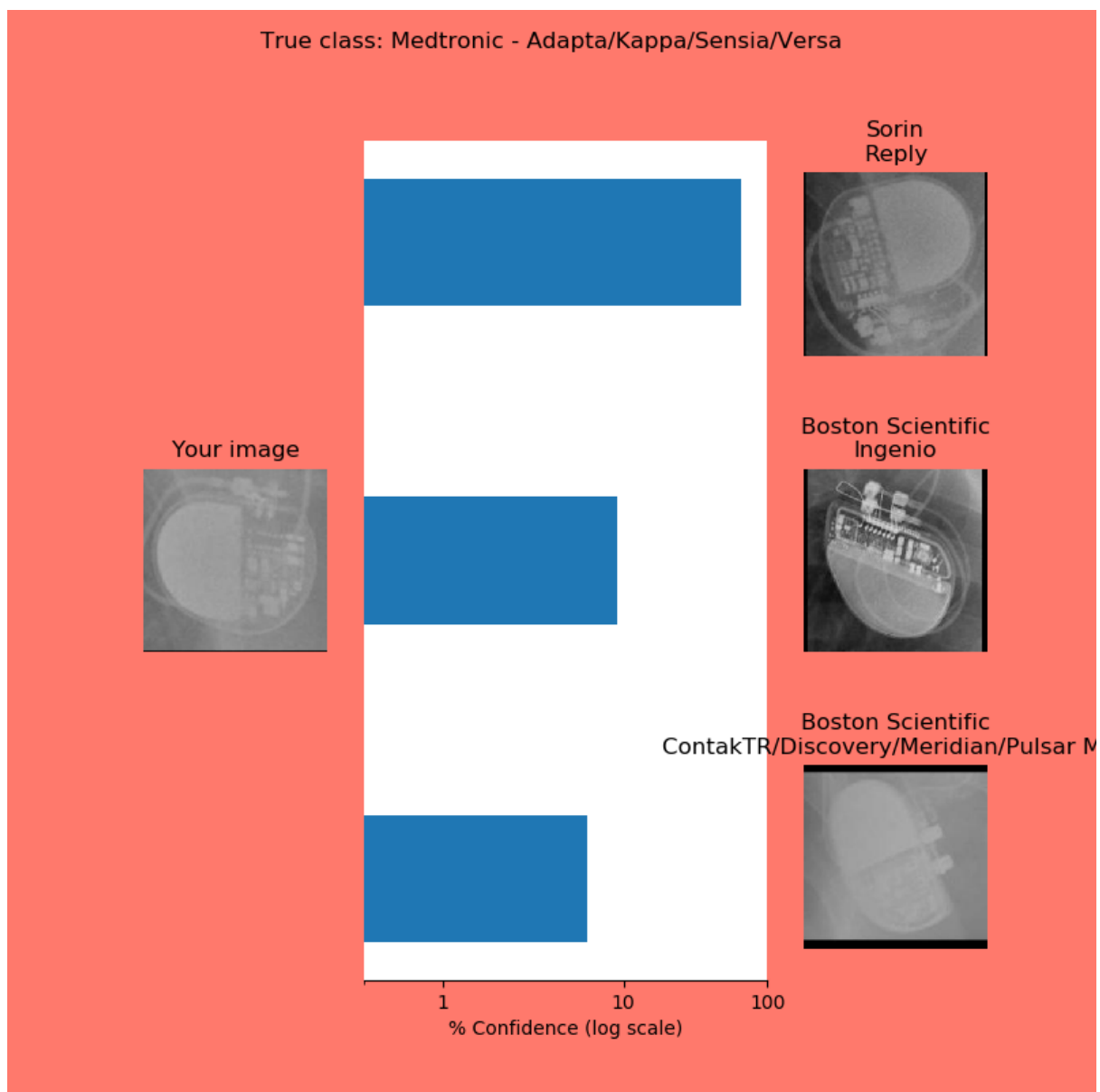


Figure 4-4 - The single case where the neural network wrongly predicted the manufacturer of a device. The device in the picture of interest (left side) is that of a Medtronic Adapta. The neural network incorrectly predicted this device to be a Sorin Reply device.

4.4.4 Comparison with human performance using the flow chart

5 human cardiologists applied the published CaRDIA-X algorithm to classify each of the 225 test set images as one of the 5 manufacturers. The accuracy of the cardiologists ranged from 62.3% to 88.9%. The median accuracy was 72.0%. The two best-performing humans were the two electrophysiologists. The median human performance was that of the best-performing non-electrophysiologist cardiologist.

The neural network was significantly more accurate than both the median (OR 63.0; 95% CI 10.9 to 2527.4; $p < 0.0001$) and best-performing human grader (OR 25.0; 95% CI 4.1 to 1026.4; $p < 0.0001$) as shown in Figure 4-5.

Neural Network Accuracy Versus Cardiologists

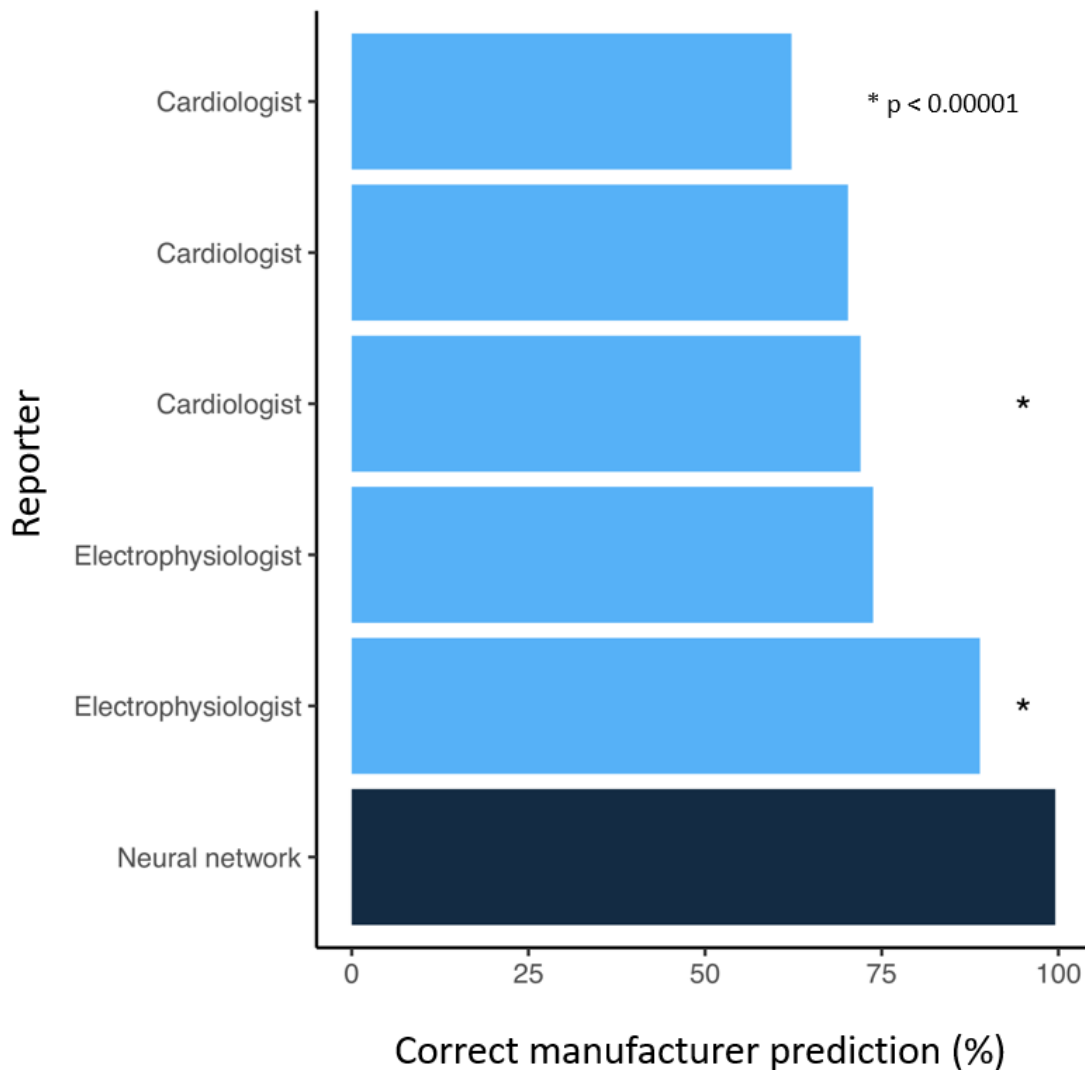


Figure 4-5 - Comparison between neural network and human performance. Bar plot showing comparative accuracy for identifying the manufacturer of devices across the five human reporters and the neural network. P values are for superiority of the neural network above the median and best human graders.

4.4.5 Visualisation of learning using saliency mapping

In an additional exploratory analysis, saliency maps were produced for each image in the test set, indicating the features of each image which are most characteristic of the pacemaker they depict. These are akin to the pathognomonic signs of a disease in clinical medicine.

To demonstrate the utility of saliency mapping, in Figure 4-6 I show 4 images comprising two different model groups. Panel A depicts a Medtronic Advisa pacemaker. I invite the examiners to identify which other panel (B, C or D) is also an Advisa, and ask how they would teach others to differentiate the two different models on an X-ray. Once they have done this, I invite the examiners to examine Figure 4-7. This figure shows that the saliency maps for the AT500 device show intense activity around a circled circuit board component which is unique to that device. I hope examiners agree that on revisiting Figure 4-6 with this knowledge, two model groups are easily differentiated.

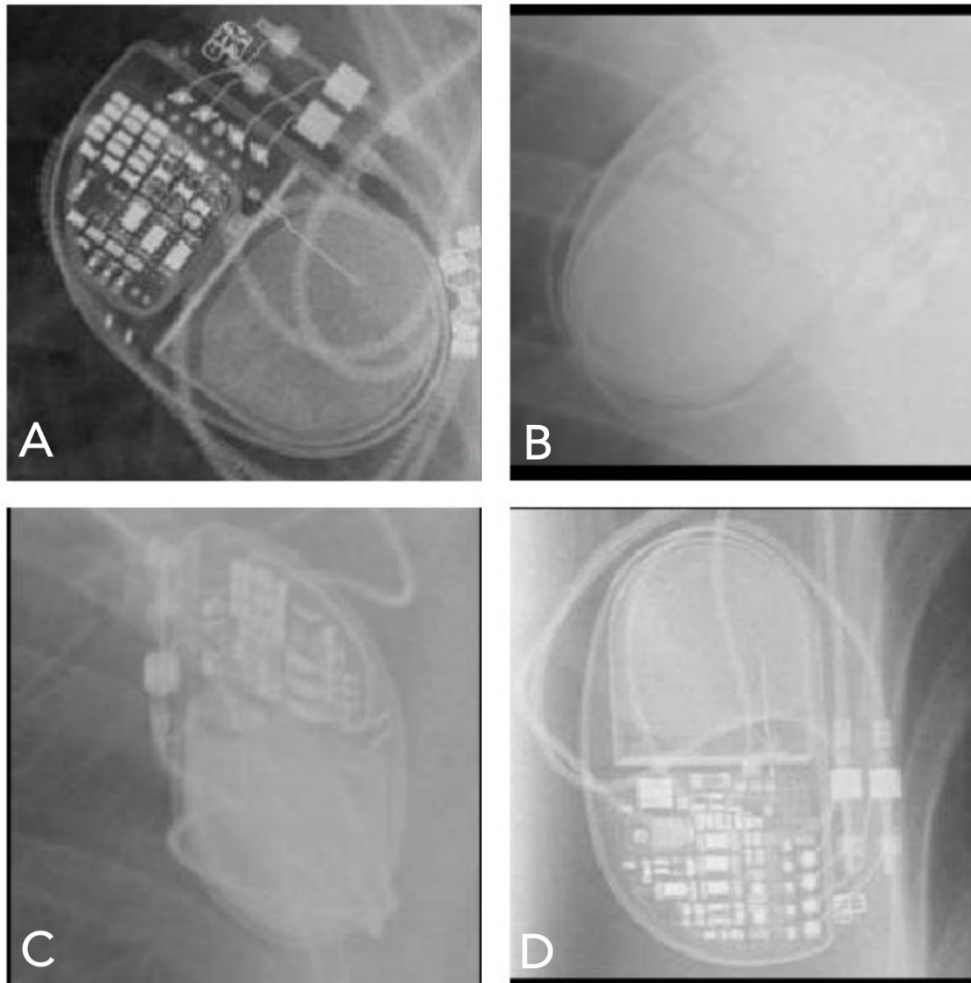


Figure 4-6 - Four images, depicting two Advisa devices, and two AT500 devices. Panel A is an Advisa. We invite readers to identify which other panel (B, C or D) is also an Advisa. The other two are AT500s. Additionally, how would you advise others to make the same distinction?

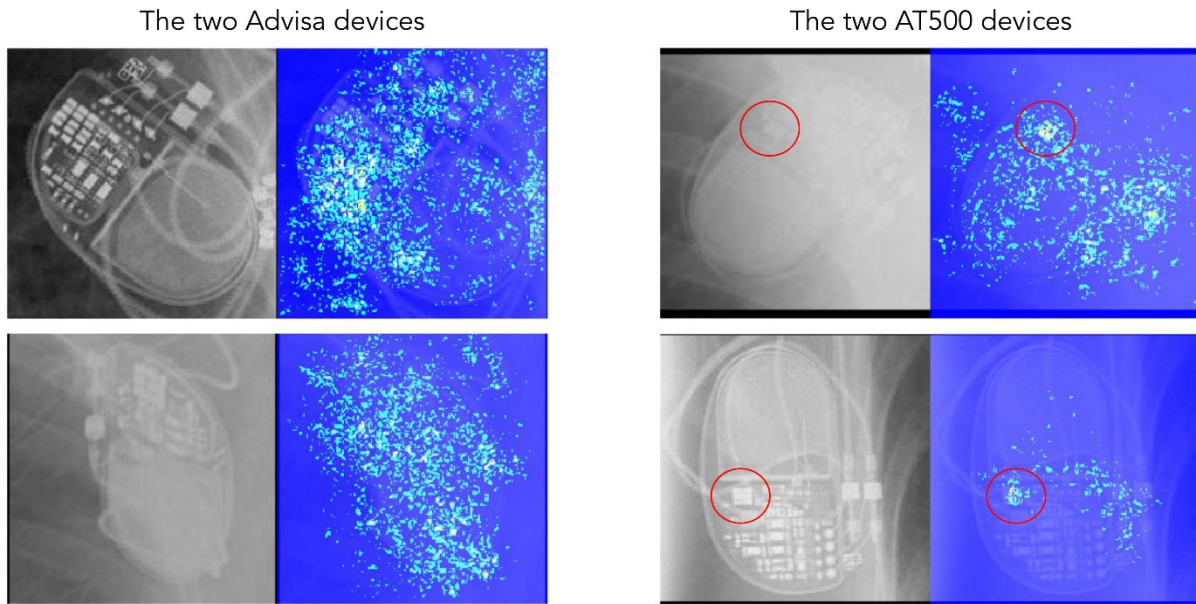


Figure 4-7 - Saliency plots from the neural network which could help guide us where to look. The answer to the question in Figure 4-6 is panel C. Saliency plots reveal that the network is focusing on a feature present in the AT500s (red circles) which is absent in the Advisa devices. Having this pointed out by the network makes it now easy to return to Figure 4-6 and correctly categorize them. These example images also demonstrate the neural network's ability to deal with dramatic differences in image quality, X-ray penetration and orientation.

4.5 Discussion

This is the first study using artificial intelligence to identify cardiac implantable electronic devices on X-rays. The convolutional neural network trained in this study delivers performance at least as good as cardiologists using a flow chart algorithm. For images which it has never seen before, the network identifies the manufacturer of the device with an accuracy of 99.6%, with corresponding cardiologist accuracies ranging from 62.3% to 88.9%. The network can also identify the specific model group with an accuracy of 96.4%.

Just like the flow chart algorithm whose use it can replace, I have made the neural network publicly and freely available for use.

4.5.1 Neural network architecture greatly affects performance

Table 4-2 shows markedly different performances across different neural network architectures. Of the neural network designs that launched machine learning into prominence, VGGNet is the only one still in common use, because of its elegant simplicity yet relatively good performance. Surprisingly, however, its performance on this task was poor. This may reflect the necessity for more advanced neural network components, such as ‘residual connections’ and ‘dimensionality reduction’ through ‘1x1 convolutions’.

ResNet was the design that pioneered residual connections, which are a method making available the original image to all subsequent layers of the network rather than only the first layer.

Separately, GoogLeNet Inception was the pioneer for condensing information between layers using 1x1 convolutions so that network’s sophistication was less constrained by the handling of vast numbers of parameters.

The design that performed best, however, was the one that made extensive use of both of these innovations, residual connections and 1x1 convolutions: Xception.

4.5.2 Clinical applications

A tool that is faster and at least as reliable as a cardiologist following a flow chart could be useful in several clinical situations. Physicians and physiologists could use it to make a quick assessment of the manufacturer of a cardiac device from a simple chest X-ray. Pacemaker programmers are portable but bulky and only the manufacturer-specific programmer will be able to communicate with the patient's device. Knowing which programmer to bring saves valuable clinical time. Not only might this facilitate rapid interrogation of a device in an emergency, but also the provision of emergency treatment, such as the delivery of anti-tachycardia pacing in a patient presenting with ventricular tachycardia.

Since I made this tool available, free to use at <http://ppm.jph.am> it has been met with international support, and several people have reported using it clinically on social media.⁷⁷

4.5.3 Saliency maps – human learning from machine learning

Machine learning has gained a reputation as ‘black box’ technology, which does not provide insights to further human understanding.^{51,52} More recently, however, saliency mapping has been developed to provide a useful window into how the neural network is making its decisions. In saliency mapping, the pixels in an image are ranked based on the strength of their effect on the network's decision.⁵³

In Figure 4-6, most humans, and indeed most expert cardiologists, have difficulty in differentiating between the two models of pacemaker. However, not only does the neural

network accurately distinguish between them, but the saliency map highlights the feature that distinguishes them most clearly. Moreover, once this salient feature is pointed out to humans (Figure 4-7), they now find it straightforward to make the distinction.

4.5.4 Study limitations

This neural network recognizes devices in common use at Imperial College Healthcare NHS Trust. However, it will not be able to identify devices not listed in Table 4-1. For example, the network at present is only trained on two implantable loop recorder devices from one manufacturer. However, the network is capable of continuous improvement. Only 25 examples of a new device are needed to train the neural network.

This study demonstrates that this neural network has superior accuracy in identifying the manufacturer of a device compared with human cardiologists and electrophysiologists using a flow-chart approach. However, we found humans performed less well on our testing dataset than reported previously.⁵ Reasons for this may include the fact the flow-chart algorithm has not been updated in 8 years, and the relative abundance of ICDs in the original dataset against which it was validated (47.1% versus 24.4% for this study), and against which the flow-chart algorithm appears to perform particularly well (see Figure 4 of the original publication).⁵

The accuracy for identifying the model group in our study is only 96.4%. Furthermore, the ‘real world’ accuracy may differ slightly from this (either better or worse) depending on the distribution of pacemaker model groups in the population. For example, if the neural network performs relatively well on the most popular model groups and relatively poorly on more rarely used devices, the accuracy may be higher than quoted in this paper. For this reason, some studies ‘weight’ their training and testing datasets by the prevalence of classes in the population to give a more accurate representation of real-world performance. Unfortunately, however, no data exist

describing the relative incidence of pacemaker models in the population, and so our dataset assumes balanced class sizes. Fortunately, in clinical practice, the most urgent question is the device manufacturer so that the correct programmer can be brought to the patient's bedside. There was only one classification error for this endpoint across the entire dataset, corresponding to an accuracy of 99.6%, which is less likely to change dramatically with the distribution of devices in the population.

Sometimes, neural networks can come to the wrong conclusion. Our website assists humans by displaying, alongside the medical staff's uploaded image, not only an image of the model that network thinks the supplied picture depicts, but also images of the two most likely alternatives. We have found that while the network's selection is correct only 96.4% of the time, in a (coincidental) 99.6% of occasions the correct model group is one of those 'top 3' displayed.

With all neural networks, there is a risk of 'overfitting'. This refers to a phenomenon where the neural network becomes excellent at recognizing images it has seen before and been trained on, but much less well on 'real-world' examples. It could be explained as the neural network 'memorising' individual images rather than actually 'learning' how to tell devices apart. We have tried to minimize the risk of overfitting in two ways. First, we have defined the network's performance as its accuracy on the 'test set' on which it was not trained. Second, we have included various methods of 'regularisation' in the network such as dropout and weight decay. These penalize the model for making decisions based on only a few very specific elements of an image, and instead favour training the model to look at a larger variety of features present.

Deployment from 'bench to bedside' can be difficult with neural networks, because the large processing power needed is not always present at the point of care. I have aimed to mitigate this by providing an online web portal that anyone can use at <http://ppm.jph.am>.

4.6 Conclusion

Machine learning and artificial intelligence are providing rapid developments in medicine, especially in the field of medical image analysis. Our approach may speed up the diagnosis and treatment of patients with cardiac rhythm devices, but I also demonstrate how neural networks are increasingly being deployed to process large quantities of medical data throughout the healthcare system, and how future patient care will likely rely increasingly on computer-aided decision-making.

Translating achievements in machine learning from ‘bench to bedside’ (or the computing laboratory to the point of care) has often proved difficult. With this study I have provided an educational online portal where physicians can interact with the network online. However, as always, further clinical studies will be essential in assessing the network’s ‘real world’ accuracy before it can be deployed as a validated clinical tool.

5 Identification of successful His Bundle Pacing using Convolutional Neural Networks

The research covered in this chapter has been previously published as:

Arnold AD & Howard JP, Gopi A, Chan CP *et al.*

“Discriminating electrocardiographic responses to His-bundle pacing using machine learning.”

Currently *In Press* at *Cardiovascular Digital Health Journal*⁷⁸

The work in this chapter was conducted jointly by myself and a colleague, Dr Ahran Arnold. Dr Arnold was responsible for the data extraction and annotation, and I was responsible for developing the segmentation software, developing the machine learning solution, and the statistical analysis. We are co-first author on the manuscript, which has been accepted for publication in late August 2020.

In contrast to the submitted manuscript, which has a strong cardiac electrophysiology focus, this PhD thesis chapter more thoroughly investigates a variety of neural network architectures, given the remarkably paucity in the literature comparing different strategies for analysis ECGs using neural networks.

The full source code I’ve developed for this project is available on my GitHub page at

<https://github.com/jphdotam/HML>

5.1 Abstract

5.1.1 Introduction

His bundle pacing is a novel alternative to traditional right ventricular pacing, where the pacing lead directly stimulates the conduction tissue of the heart. Depolarisation patterns are therefore typically more physiological than traditional pacing and may be associated with lower morbidity and mortality. However, successful His bundle pacing requires staff to be able to discriminate between three electrocardiographic responses to pacing: (1) selective His bundle pacing (S-HBP), (2) non-selective His bundle pacing (NS-HBP), and (3) myocardium-only capture (MOC). This classification can be time consuming and is reliant on precise measurements, which can be difficult for those new to the field.

We set out to identify whether a convolutional neural network could successfully classify these three electrophysiological responses to His bundle pacing.

5.1.2 Methods

Raw 12-lead electrocardiograms were extracted from successive patients undergoing His bundle pacing at a single centre. Paced beats were extracted, segmented using custom made software, and classified as showing either S-HBP, NS-HBP or MOC. 75% of cases were used to train a series of 1-dimensional convolutional neural networks, and 25% of cases were used as a final testing set.

5.1.3 Results

The training set comprised 1297 ECGs from 59 patients. The testing set comprised 140 ECGs from 17 patients. The best-performing neural network correctly classified 105 (75%) of the

ECGs (Cohen's Kappa 0.59, 95% CI 0.30 to 0.88, $p < 0.0001$). Accuracy was 67%, 71% and 84% for S-HBP, NS-HBP and MOC ECGs, respectively.

5.1.4 Conclusions

This proof of concept study demonstrates that a 1-dimensional convolutional neural network is able to classify electrocardiographic pacing responses during His bundle pacing. This performance is without supplying the network with the baseline ECG parameters or performing complex pacing manoeuvres, which manual algorithms currently rely on. Larger datasets may further improve this performance, allowing artificially intelligent ECG analysis to assist with His bundle pacemaker implantations and follow-up.

5.2 Introduction

5.2.1 His bundle pacing is a novel form pacing

Right ventricular pacing (RVP) is the traditional method of cardiac pacing, where pacing is mediated through a pacing wire introduced into the right ventricle. However, the resultant depolarisation pattern is abnormal: instead of the left ventricle being depolarised synchronously through the heart's native conduction system, dyssynchronous depolarisation and contraction occurs as depolarisation propagates slowly via cell-to-cell activation.⁸ This results in a left bundle branch block (LBBB) appearance on the ECG and appears to have a detrimental effect upon cardiac function, with long-term RVP associated with ventricular impairment, heart failure and mortality.⁹

Biventricular pacing (BVP) was therefore developed as an alternative to RVP.⁷⁹ This involves introduction of a second pacing wire, typically via the coronary sinus, allowing the lateral wall of the left ventricle to be paced simultaneously with the right ventricle.⁸⁰ However, despite biventricular pacing sometimes being referred to as cardiac resynchronisation therapy (CRT), there still exists residual dyssynchrony as each ventricle still depolarises through cell-to-cell conduction from the pacing site.¹¹

His bundle pacing (HBP) is a novel form of pacing where the pacing lead activates the intrinsic conducting system of the heart: the His bundle.¹⁰ This aims to provide physiological activation of the ventricles. Indeed, it appears successful HBP can “reverse” bundle branch blocks, restoring a normal QRS ECG pattern.^{81,82} It is thought that this may allow HBP to overcome the detrimental effects of long-term RVP, and provide an alternative to BVP in patients with heart failure.⁸³

5.2.2 Successful His bundle pacing requires skilled ECG interpretation

The ECG resulting from attempted His bundle pacing can take on several patterns, depending on whether pacing captures the His bundle (selective His bundle pacing), local myocardium (myocardial only capture), or both (non-selective His bundle pacing).

Selective His bundle pacing (S-HBP) occurs when pacing captures the His bundle alone, with no capture of the surrounding myocardium. The ECG during S-HBP will demonstrate a pacing spike, an isoelectric segment during which the depolarisation propagates through the conduction system, and then the QRS complex. In patients with underlying normal QRS morphology, the appearance of the QRS complex remains unchanged. In patients with bundle branch block, the extent to which the QRS normalises (termed bundle branch block correction) depends on the location of the block: block in the proximal conduction system can be overcome by HBP, but block in the distal Purkinje fibres cannot.

Non-selective His bundle pacing (NS-HBP) occurs when pacing captures both the His bundle and the surrounding myocardium. As with S-HBP, depolarisation of the ventricles is mediated by the native conduction system, and the QRS morphology will appear similar to the native QRS, or even normalise in some patients with bundle branch block. However, capture of the local myocardium results in immediate depolarisation of the basal right ventricular septum, and so a slurred QRS onset is seen immediately following the pacing spike, rather than the isoelectric segment seen in S-HBP. This is sometimes termed a “pseudo delta wave”.

Myocardium only capture (MOC) represents a failure in His bundle pacing, where the His bundle is not captured. As with NS-HBP, the right ventricular septum is captured immediately following the pacing spike, and indeed MOC is effectively right ventricular septal pacing. The QRS can appear similar to NS-HBP because of the lack of isoelectric segment, and discriminating between these two appearances is particularly difficult in patients with underlying LBBB who display only partial, or no, correction.

Accurately identifying these three responses to attempted His bundle pacing therefore requires considerably more expertise in rapid ECG interpretation than is required during traditional pacing procedures. Nevertheless, it is of crucial importance: the inability to discriminate between NS-HBP and MOC in patients represents an inability to identify whether His bundle pacing has been achieved.

An automated system to discriminate between these three types of capture is therefore highly desirable. However, a successful system would need to involve multiple steps, if programmed using a traditional stepwise algorithmic approach. These would include automated identification of the interval between the pacing spike and the QRS onset, analysis of this interval across the 12 leads to ascertain whether it is isoelectric, identification of the QRS complex, and finally QRS morphology assessment during pacing in the context of the patient's native QRS. Furthermore, this process would need to work across a variety of baseline ECG appearances including narrow QRS, LBBB and right bundle branch block (RBBB), and be resilient to noise across all or a proportion of the ECG leads.

We therefore set out to investigate whether a system using artificial intelligence (AI) could learn how to perform this task, without the need for programming a manual algorithmic solution. I have had success in the past using convolutional neural networks (CNNs), a form of artificial intelligence, to classify biological waveform data,¹³ and therefore I attempted to see whether CNNs could be used to discriminate between the three responses to attempted His bundle pacing.

5.2.3 Convolutional neural networks for classifying biological waveforms

This study differs from the other chapters of my thesis, in that the CNNs employed are 1-dimensional (1D). This may appear confusing, as technically the ECG data itself is 2-dimensional. However, the underlying data for each lead is a single stream of numbers which is plotted as a graph against time. Therefore, the ECG signal is a 1-dimensional data-stream for each lead. Therefore, the *convolution* is 1-dimensional: the filters start at the beginning of the trace and slide forward through time, in 1 dimension. This is analogous to images: technically colour image data could be view as three-dimensional (a height in pixels, a width in pixels, and a depth of 3 ‘channels’ storing the quantity of red, green and blue in each pixel), but the convolution only occurs in two dimensions (height and width).

In comparison with 2D CNNs, 1D CNNs have been the focus of relatively little research. This is in part due to image analysis being an inherently larger field than waveform analysis, with more consumer applications (facial recognition, optical character recognition, identifying explicit content on social media etc.). However, there are two other reasons that 1D CNNs have received comparatively little interest.

First, traditional non-AI-based algorithmic approaches have already been successful in the classification of waveforms. For example, for decades clinical ECG machines have been capable of providing some automated interpretation, providing estimates of the P-R interval, QRS duration, and even detection of abnormal features such as ST deviations and T wave inversion.⁸⁴ This is in stark comparison with medical image data; only in the last few years have systems been developed that have allowed assisted chest X-ray interpretation,³² and these systems are still not in routine clinical use.

Second, there exists an alternative neural network method to 1D CNNs for waveform classification: recurrent neural networks (RNNs). Multiple types of RNNs exist, but two popular types, long short-term memory networks (LSTMs) and gated recurrent units (GRUs) work by being fed every element of a sequence in turn. They contain multiple ‘cells’, each containing memory banks and circuits allowing them to learn what things should be ‘remembered’ and what should be ‘forgotten’. RNNs are significantly more complex in structure than 1D CNNs, and have been shown to excel at many traditionally difficult tasks, such as speech recognition⁸⁵ and text translation⁸⁶.

Previous groups have employed both RNNs^{87,88} and CNNs⁸⁹ to classify ECGs, and in this paper, I will investigate both approaches.

5.3 Methods

5.3.1 Identification of cases

We identified patients undergoing HBP for any indication with continuous 12-lead ECG recording at Hammersmith Hospital between 2014 and 2020. No limitations were set on the baseline QRS morphology, with patients with narrow QRS complexes, LBBB and right bundle branch block (RBBB) eligible for inclusion. Patients with bundle branch block were eligible regardless of the degree of correction with HBP. Patients were eligible for analysis if they contributed at least 5 beats for at least one of the three beat classes (S-HBP, NS-HBP or MOC). ECGs were randomly assigned at the patient level between the training and testing datasets in a 3:1 ratio.

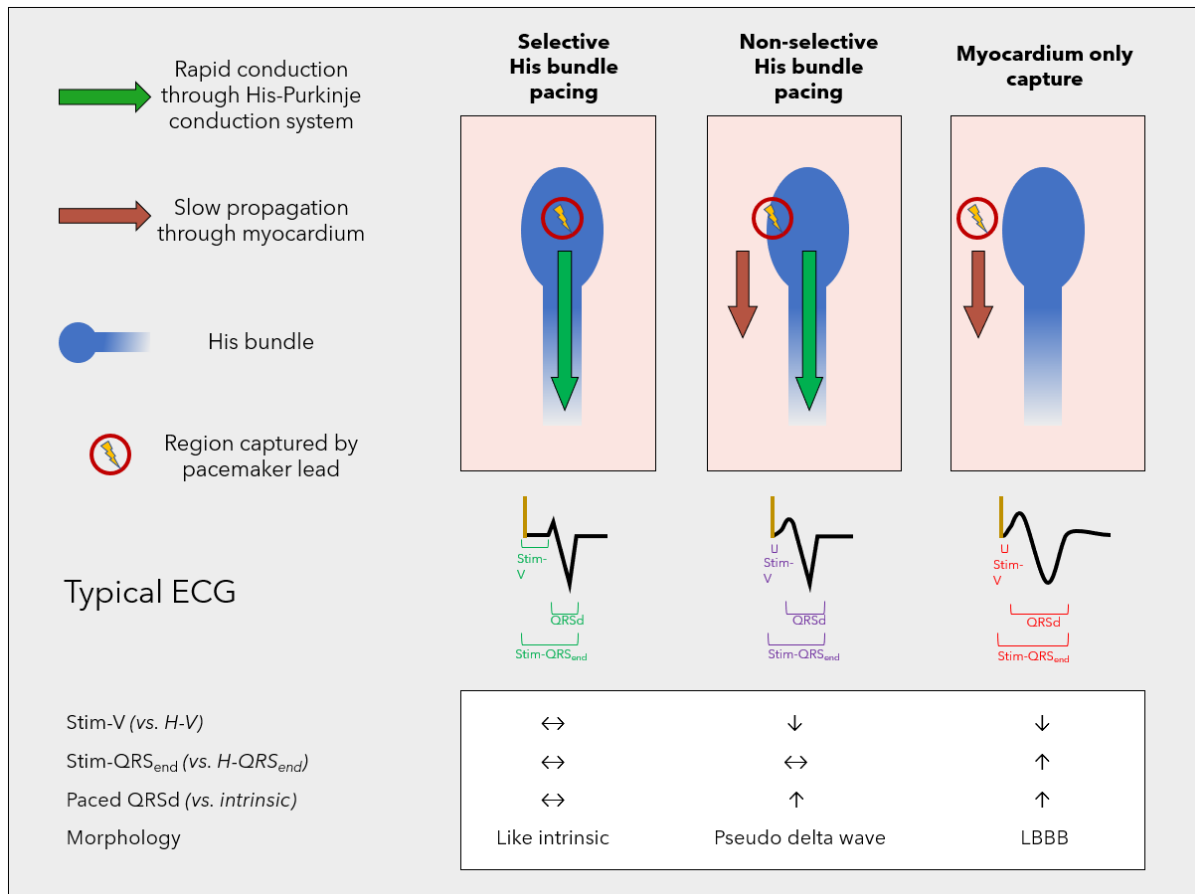


Figure 5-1 - The mechanisms and ECG findings of the three capture types seen during attempted His bundle pacing. Stim-V = interval between start of pacing spike and start of QRS; Stim-QRS_{end} = interval between start of pacing spike and end of QRS; QRSd = duration of QRS complex. Adapted with permission from Arnold & Howard et al.⁷⁸

5.3.2 Data extraction and dataset creation

We extracted digital electrocardiograms (ECGs) at a sampling frequency of 1000 Hz. Up to 50 paced beats per case were extracted by medical students who were trained to identify pacing manoeuvres recording during the electrophysiology cases, where were then labelled as one of the three beat types by an expert (see Section 5.3.3).

The ECGs were filtered using a wavelet filter using a Coiflets 5 wavelet. Only data falling between the end of a pacing spike and the end of the QRS-T complex were extracted (see Section 5.3.4). For cases in the training dataset, all beats were used to train the network.

However, for cases in the testing dataset, only 5 beats per class per case were used to assess the neural network's performance. These 5 beats were selected randomly. This was to ensure balanced number of unique ECGs for the final analysis and prevent cases with longer recordings contributing a disproportionate number of traces to the testing dataset, biasing results.

Cases were randomised in a 3:1 ratio between the training dataset (used to develop and train the neural networks) and testing dataset (used to report the neural networks' final performances) – see Figure 5-5.

The training dataset was used to the neural networks in a process termed 'three fold cross-validation' (Figure 5-5). This is performed by splitting the training dataset into thirds, termed 'folds'. Then three neural networks of the same design are created, and each one is trained using two thirds of the training data, and its performance tracked (validated) using the remaining third. The third used for validation is different for each of the three trained networks. This allows each of the cases in the training set to contribute towards network training, whilst still allowing the neural network's performance to be adjusted during development on validation data independent

of the final ‘testing’ set. During testing, the predictions of the three networks are averaged to provide a final ‘ensembled’ prediction.

5.3.3 Classification of beats

Dr Ahran Arnold, an expert in HBP ECG analysis, classified 12-lead ECG beats as showing either S-HBP, NS-HBP or MOC using visual inspection and electronic calliper measurements. His bundle lead electrograms were also used where available. The criteria used is shown in the online appendix of the published manuscript, but are also outlined in Figure 5-1, and followed international recommendations.¹²

For this proof of concept study, we excluded rare forms of capture, which included isolated bundle branch capture or fusion, as insufficient numbers of beats were present to allow us to train the classifier on these categories.

5.3.4 Segmentation of beats

I wanted to ensure the neural network was able to identify beats using only the QRS complex of interest, and not adjacent information which may bias it. For example, MOC is commonly encountered at lower pacing energies, and NS-HBP at higher pacing energies. Indeed, many of the MOC beats we extracted were taken from ‘threshold checks’, where the output energy of the pacemaker was reduced until an NS-HBP pattern reverted to an MOC pattern. A crude classifier may therefore learn to rely purely on the amplitude of the pacing spike to discriminate between these beats. Such a classifier might perform well on our testing set, but would be unsuitable for any clinical use.

I therefore developed an interface which would allow us to manually segment out the pacing spike, QRS, RR interval and T waves of an ECG. This software has a labelling interface (Figure 5-2) and a data export interface (Figure 5-3). The full source code for this software is available on my GitHub at <https://github.com/jphdotam/BARD-labeller>.

Following this, I was able to extract ECGs that fell between the end of the pacing spike and the end of the QRS-T complex, excluding the pacing spike itself (see Section 5.3.2).

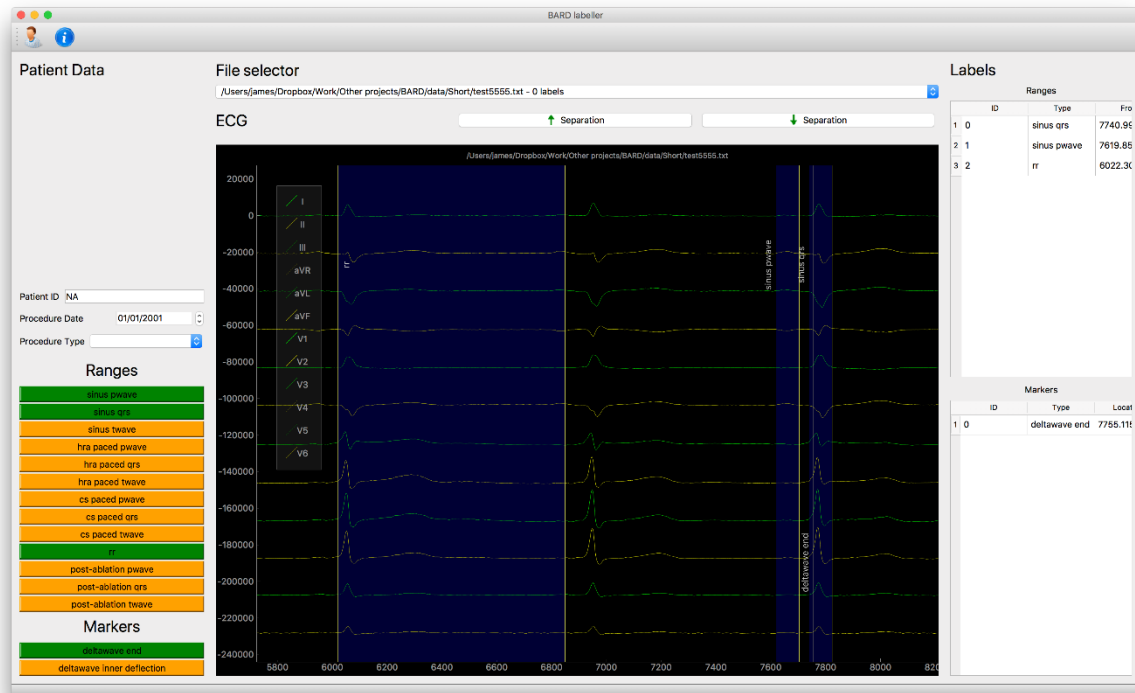


Figure 5-2 – The segmentation interface of the software I developed to allow manual segmentation of individual ECG beats.

The buttons on the left hand side indicate whether a segment of the ECG has been segmented (green) or not (amber).

Segmentation is done using blue slides which can be stretched over a region of interest of the ECG trace.

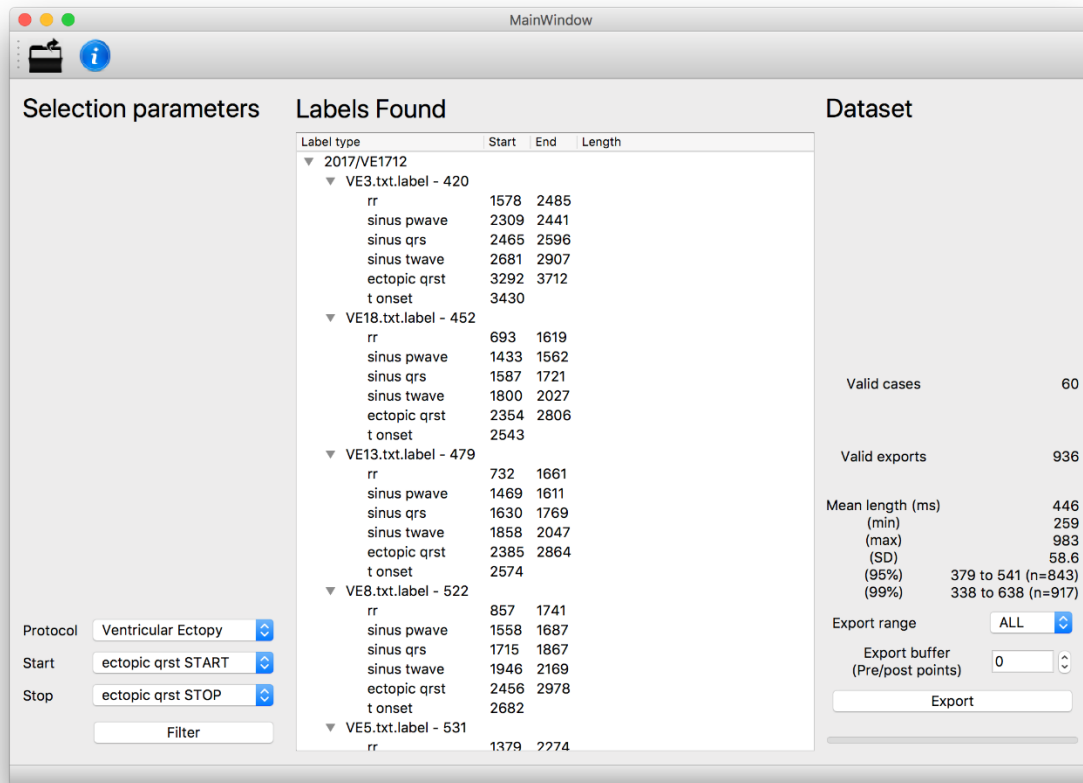


Figure 5-3 - The data export interface of the software I developed to allow batch exporting of labelled ECGs. Certain protocols and start/stop boundaries can be chosen in the bottom left corner, to allow specific areas of specific ECGs to be extracted. The exported data will then comprise a database from which a neural network can be trained.

5.3.5 Neural network development

In this study, I investigated several neural network architectures. I created all of the neural networks using the Python programming language (Python Software Foundation, Delaware, United States) and the Pytorch machine learning framework.⁶⁴

5.3.5.1 Re-implementation of the state-of-the-art convolutional neural network

First, I implemented the 1D CNN used by Hannun *et al.* to classify ECGs in *Nature Medicine*.⁸⁹ As no source code for this network was available, I constructed a network according to the diagrams and descriptions in the index publication. Some aspects of their design were therefore unclear, as outlined below:

First, Hannun *et al.* describe down-sampling in alternating blocks, but did not explain the method used to accomplish this. I therefore implemented this using strided convolutions.

Second, when down-sampling was performed within a block, it was not clear which of the two convolutional layers did this. In my implementation, the stride was performed by the second convolutional layer.

Third, when the number of filters in a layer increased, it was unclear how the residual connection bypassing the layer could account for this. I therefore employed a single convolutional layer with a batch normalisation layer in place of the standard residual connection (the strategy used in the ResNet networks).

Finally, as their network was designed to accept long strips of ECGs lasting up to 30 seconds, as opposed to the single ECG complexes in this study, I reduced the filter width from 16 to 3 (as used by the ResNet architecture).

5.3.5.2 The creation of novel 1-dimensional convolutional neural networks, adapted from established 2-dimensional convolutional neural networks

I went on to adapt several previous state-of-the-art 2-dimensional neural network families to a 1-dimensional design.

The ResNet family includes a variety of neural network architectures, comprising between 18 and 101 layers.⁷¹ I assessed each of these separately. Two prototypical examples, ResNet34 (panels A&B) and ResNet50 (panels C&D), are shown in Figure 5-4.

ResNet50 is the smallest ResNet that makes use of “bottleneck” blocks (Figure 5-4). These swap out the traditional approach of using two convolutional layers of size 3, and instead use three convolutional layers of size 1, 3 and 1, respectively. This allows deeper networks with more layers to be created, whilst minimising the numbers of parameters (and therefore memory and compute requirements) of each layer.

The WideResNet⁹⁰ and ResNeXt⁹¹ architectures were successors to ResNet. The WideResNet architecture is almost identical to the ResNet architecture, except each convolutional block is “wider” (containing more filters per layer). ResNeXt is also similar to ResNet, but each residual block contains multiple parallel paths through which the data flows, with fewer filters, rather than a single channel with more filters. These changes manifest in the bottleneck layers, and so WideResNet and ResNeXt are only exist as variants of ResNet50 or larger. Both of these networks have been shown to be superior to ResNet in image classification examples.

Finally, DenseNets are a separate architecture, inspired by (but fundamentally different from) ResNets. Whereas ResNets summate (add) the output from previous layers sequentially using skip connections, DenseNets concatenate (join) them.

The WideResNet, ResNeXt and DenseNet architectures have been shown to be superior to ResNets in image classification examples^{72,90,91}, and in this chapter I have adapted them to 1D CNNs and appraised their performance in this ECG classification task.

5.3.5.3 Re-implementation of the state-of-the-art RNNs

Finally, I implemented two previously validated recurrent neural network designs (RNNs). The two RNNs are re-implemented from the paper by Singh *et al.*⁹² and comprise a network using multiple long short term memory layers and a network using multiple gated recurrent unit layers.

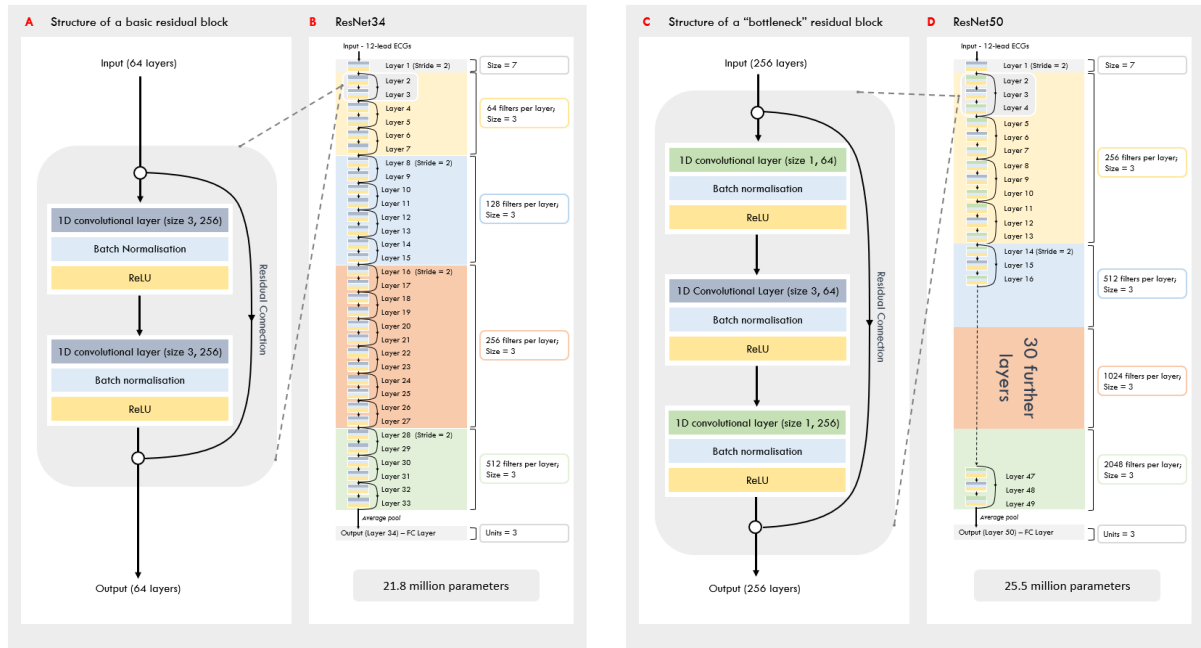


Figure 5-4 - Two examples of ResNet networks. Panels A&B show ResNet34. This network comprises 34 layers (panel B), which is made up of a series of “basic blocks” (panel A), which are two sets of convolutional layers of size 3 (dark blue), a batch normalisation layer, and a rectified linear unit (ReLU). Importantly, there is a “skip” or “residual” connection across the two layers, which allows data to bypass the layers if this is advantageous during training. Larger ResNet networks, such as ResNet50 (panels C&D), use a different building block, termed a “bottleneck block” (panel C), which comprises three layers. The first and third layer are convolutional layers of size 1 (green) rather than 3. This allows more layers to be used, allowing the network to be deeper, without the number of weights within the networks being prohibitively large.

5.3.6 Neural network training

To train the neural network, I fed in batches of 32 ECGs at a time. Shorter ECGs were padded with zeros to ensure every ECG within a batch was the same length. The ECGs were normalised by zeroing each lead to its first datapoint, and dividing by the lead-wise IQR across all cases. The loss was calculated over each batch using the categorical cross entropy loss function with label smoothing ($\alpha=0.1$) and online hard example mining (rate=0.9). Class weights were used (5:1:1 for MC, S-HBP and NS-HBP, respectively), to help account for class imbalance. Weights were updated with the AdamW optimizer with a learning rate of 0.001 and weight decay of 0.01.

The neural networks were trained using two RTX Titan graphical processing units (Nvidia Corporation, California, United States). I have made the source code available online at <https://github.com/jphdotam/HML>.

5.3.7 Visualisation of learning

I used saliency mapping (see Section 2.10, page 90) to try to identify which aspects of the ECG traces were (1) contributing to correct classifications, (2) contributing to incorrect classifications. This was performed in a similar way to training the network, with an extra step. When we train a CNN, we feed in data and calculate how the all the trainable parameters (weights) in the neural relate to the correct answer, so we can adjust them in a favourable way. In this task of saliency mapping, I extended this process back one step further, beyond the first layer of the network, to the original ECG: I saw which sections of the ECG were most strongly linked to the neural network's opinion. Saliency maps for every testing set example were analysed qualitatively by Dr Ahran Arnold, a specialist in HBP interpretation.

5.3.8 Statistical analysis

I assessed the performance of the neural network using Cohen's Kappa. Cohen's Kappa, unlike the simple accuracy statistic, accounts for imbalanced class sizes. For example, imagine a classifier such as a neural network which was designed to identify images of cats and dogs. If 80% of the dataset is dogs, then the classifier could attain 80% accuracy by merely predicting dog for every image, regardless of its content. However, the Cohen's Kappa for such a classifier would be 0, indicating the performance of the classifier is no better than would be expected due to chance, given the proportion of each class in the dataset. The maximum value for Cohen's Kappa is 1.0, indicating a perfect classifier.

5.4 Results

5.4.1 Study dataset

A total of 84 cases were eligible for analysis (Figure 5-5).

59 cases, comprising 1297 ECGs, were assigned to the training dataset. These cases were split into three folds, which were used to train 3 neural networks of each architecture using three-fold cross validation.

17 cases, comprising 140 ECGs, were assigned to the testing dataset. This dataset was used to assess the final performance of each architecture. Fewer ECGs per patient were used for the testing dataset, to ensure a balanced number of unique ECG appearances across cases (see section 5.3.2, page 143).

The baseline characteristics for the two datasets are shown in Table 5-2.

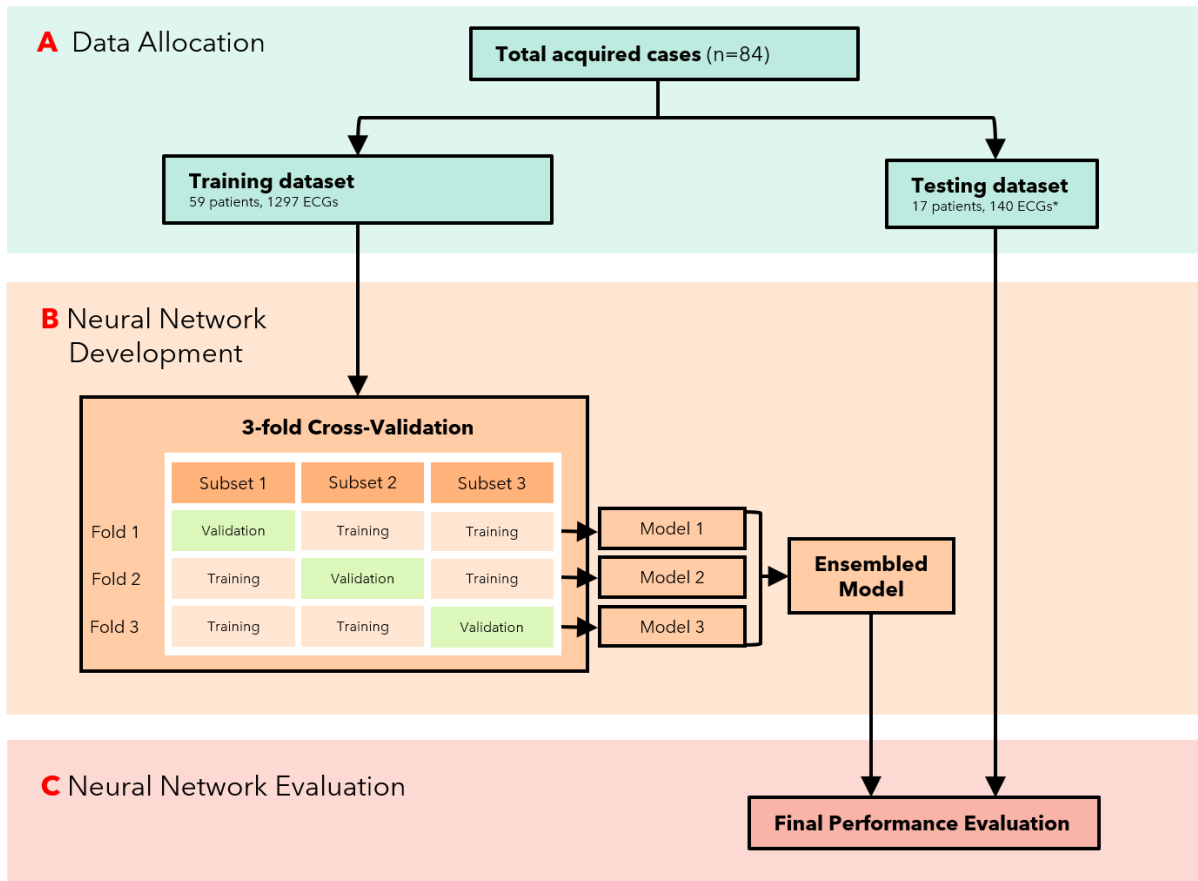


Figure 5-5 - Study flowchart. Individual patients were randomised between the training and testing datasets, with the ECGs extracted from each case. The training dataset was further divided into 'folds', with three neural networks trained for each architecture. *Fewer ECGs per patient were used for the testing dataset, to ensure a balanced number of unique ECG appearances across cases (see section 5.3.2, page 143).

	Training set (n=59)	Testing set (n=17)
Age (years)	72 ± 11 (40-84)	75 ± 11 (47-86)
Male	63 (83%)	14 (82%)
Ischemic heart disease	31 (53%)	7 (41%)
Ejection fraction (%)	35 ± 10 (12.5-50)	35 ± 11 (14-52.5)
Indication		
CRT	22 (37%)	9 (52%)
1° AVB	22 (37%)	2 (12%)
Sinus node dysfunction	11 (19%)	0 (0%)
High degree AV block	4 (7%)	6 (35%)
Underlying rhythm		
Sinus rhythm	53 (90%)	14 (82%)
Atrial fibrillation	4 (7%)	2 (12%)
High degree AV block	2 (3%)	1 (6%)
Underlying QRS morphology		
Normal	24 (34%)	4 (24%)
LBBB	20 (26%)	8 (47%)
RBBB	15 (41%)	6 (35%)
Cases including each ECG morphology*		
S-HBP	37 (58%)	10 (35%)
NS-HBP	46 (32%)	15 (54%)
MOC	19 (18%)	3 (11%)
Total ECGs**	1297	140

Table 5-1 - Patient characteristics. Values are mean ± standard deviation (range) for age and ejection fraction. Otherwise n and percentage are given.

* Percentage of total number of case types.

** Cases composing the testing set were limited to contributing exactly 5 beats of each morphology to ensure accuracy measurements were balanced across patients

5.4.2 Performance of the different neural network architectures

The neural network architectures assessed are shown in Table 5-2. Cohen's Kappa ranged between -0.207 for the worst performing network (the long short-term memory-based recurrent neural network, accuracy 37.7%) to 0.588 for the best performing network (ResNet34, accuracy 75.0%).

The re-implementation of the network used by Hannun *et al.* in *Nature Medicine* performed significantly poorer than ResNet34 (Kappa 0.427 versus 0.588; $p < 0.001$).

The confusion matrix for the best-performing neural network (ResNet34) is shown in Figure 5-6. Once trained, this neural network took 0.08s to process each beat.

Architecture	Parameters (millions)	Accuracy (%)	Kappa
State of the art convolutional neural network			
HannunNet⁸⁹	2.1	69.6	0.427
ResNet			
ResNet18	1.0	72.6	0.498
ResNet34	1.8	75.0	0.588
ResNet50	4.0	64.4	0.371
ResNet101	7.1	63.7	0.351
ResNet152	9.7	63.0	0.338
Wide ResNet			
Wide ResNet50	9.2	69.6	0.460
Wide ResNet101	17.0	68.9	0.427
ResNeXt			
ResNeXt-50	5.5	73.3	0.527
ResNeXt-101	20.0	71.1	0.491
DenseNet			
DenseNet-121	5.5	68.9	0.439
DenseNet-169	22.1	65.2	0.300
DenseNet-161	10.5	71.9	0.486
DenseNet-201	15.7	63.0	0.285
State of the art recurrent neural networks			
RNN GRU⁹²	0.2	60.7	0.367
RNN LSTM⁹²	0.2	37.7	-0.207

Table 5-2 - Results on the testing dataset for the different neural network architectures assessed.

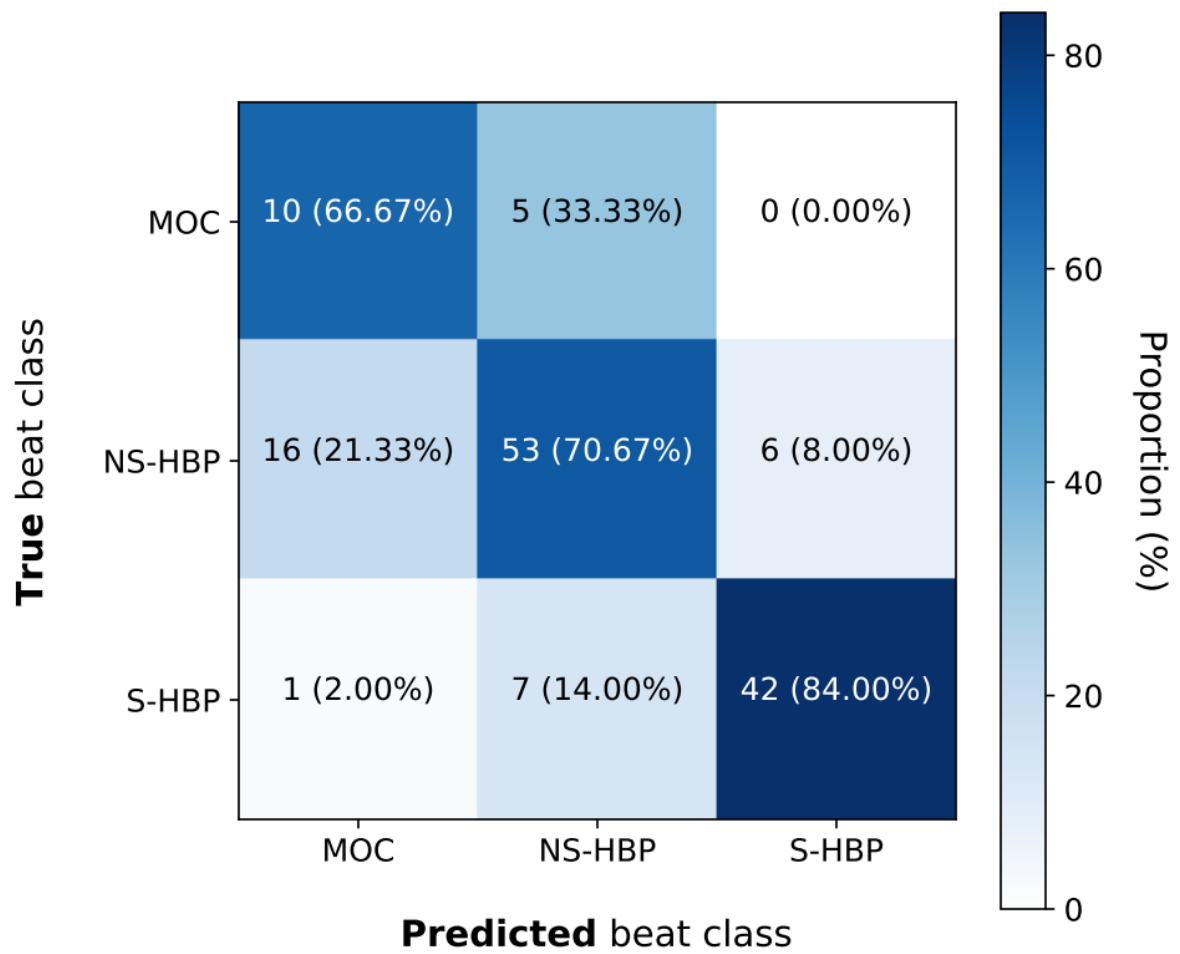


Figure 5-6 - Confusion matrix for the best performing neural network. ResNet34 demonstrated 75% accuracy (kappa 0.588) in classifying beats across the three classes; MOC = myocardial-only capture; NS-HBP = non-selective His bundle pacing; S-HBP = selective His bundle pacing.

5.4.3 Saliency mapping

Figure 5-7 demonstrates 4 panels showing saliency maps produced from 4 testing dataset ECGs. Only the 6 chest leads are shown for clarity, but the network analyses all 12 leads.

Panel A demonstrates an example of S-HBP which was correctly diagnosed by the neural network. This showed the iso-electric interval appears to be salient, which is also the key feature used for human expert analysis in discriminating between S-HBP and NS-HBP.

Panel B demonstrates an example of NS-HBP which was correctly diagnosed by the neural network. The pseudo-delta-wave (section 5.2.2, page 137) appears to be salient, which is the key feature used for human expert analysis in discriminating between NS-HBP and S-HBP.

Panel C demonstrates an example of MOC which was correctly diagnosed by the neural network. Slurred early activation and slow late activation are salient, and both are key features for human expert analysis.

Panel D demonstrates an example of NS-HBP which was **incorrectly** diagnosed by the neural network. This particular kind of ECG analysis is also difficult for human experts, and involves the analysis of behaviour at different pacing thresholds and the intrinsic QRS morphology, neither of which the neural network can access.

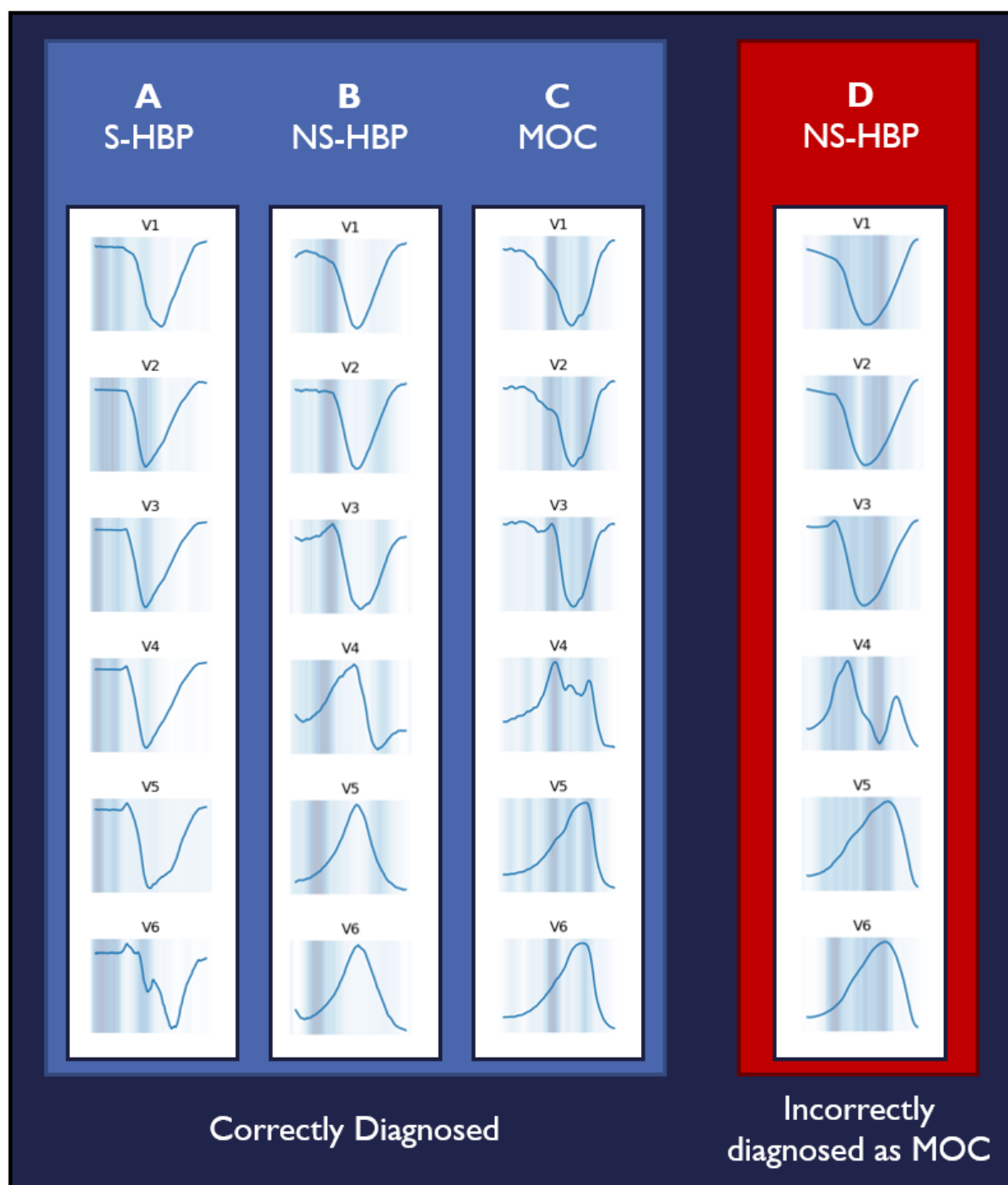


Figure 5-7 - Saliency maps demonstrating activity for chest leads. Dark blue areas are more salient. This is an indication of which parts of the ECG are contributing most towards the network's decision.

5.5 Discussion

5.5.1 Clinical significance

In this study we have shown that a neural network can discriminate between the different ECG responses to His bundle pacing. This was in the context of a relatively small dataset of only 82 patients and significant heterogeneity of beats within each class (with varying degrees of bundle branch block seen). The model's performance was imperfect, however, with 75% accuracy, and there was still a 16% error rate even in the most accurately diagnosed category of beats (S-HBP).

However, this system offers numerous advantages despite its limitations.

First, human expert classification is difficult and currently requires comparing paced beat with the patient's intrinsic pre-procedure rhythm. It also requires manual measurements of ECG intervals and often pacing manoeuvres to elicit changes in the QRS morphology using a pacemaker programmer. In comparison, this system uses a single paced beat. For this reason, the system is fully automatable, as no selection of baseline beats or calliper placement is required.

Second, this system is fast. The CNN processes beats in less than a tenth of a second. This means that whilst this system's accuracy remains imperfect, it could prove useful as a beat-to-beat screening system during His bundle pacing. During long periods of adjusting a pacing wire, the system could provide continuous assessment of the ECGs being recorded, providing real time feedback of any perceived changes in His bundle capture.

Third, this system is a proof of concept, and we plan on continually improving its performance. We have open sourced the source code online, and as our centre performs more cases, we will continue to refine the network through further training. We may also be able to introduce more advanced features. For example, I have previous experience using neural networks where a 1D CNN analyses individual beats, and another neural network 'wraps' around the original CNN, allowing successive beats to be analysed.¹³ I could use such a system to train a network to analyse

beats in the context of increasing voltages, so the outer network can make a decision based on how the features of the ECGs change.

5.5.2 CNNs outperform RNNs at classification of ECG responses to

His bundle pacing

In this Chapter I have demonstrated that every one of the 14 convolutional neural network designs I assessed were superior to the recurrent neural network using the long-short term memory blocks, previously used to classify ECGs in Singh *et al.*'s 2018 paper.⁹² Furthermore, 10 of the 14 CNN designs were superior to the RNN using the gated recurrent unit blocks in terms of Cohen's kappa.

This matches my previous experiences of classify medical waveform data (aortic pressure traces) using neural networks, I have found 1D CNNs to be superior,¹³ and indeed, a recent paper in *Nature Medicine* by Hannun *et al.* used a 1D CNN with no RNN components to identify classify ECGs.⁸⁹

I suspect the recent shift away from RNNs to CNNs for waveform classification is due to the inherent constraints on CNNs, dictating what they can and cannot learn. As discussed above, a CNN can only learn by developing and matching small templates (filters) with the input data (an ECG). Typically, the number of filters in the first layer of a CNN is much smaller than the number of training examples (e.g. 64 filters for the training set of over 1000 ECGs in this paper¹³). This means that the CNN must use the limited number of filters efficiently, identifying basic patterns that are common across the hundreds or thousands of ECG traces. The CNN *cannot* use a unique filter for identifying each ECG, in an attempt to “memorise” each one – this would be a form of overfitting (Section 2.9, page 84). RNNs are not subject to these spatial constraints of CNNs.

5.5.3 ResNet34 is superior to other network architectures, including larger ResNets and those derived from them

The best performing CNN out of the 14 I assessed was ResNet34. This network performed better than several other neural networks which have been shown in other fields to offer performance advantages, and I will explore these here.

First, ResNet34 was the best performing of the 5 classical ResNet architectures (ResNet18, 34, 50, 101 and 152). The superiority of ResNet34 over ResNet18 is likely attributable to ResNet34's greater 'depth'; ResNet34 comprises 16 more layers than ResNet18, and so has a greater ability to learn and process data. However, I have shown that deeper ResNets (50, 101 and 152) display markedly worse performance than ResNet34 in this study. I believe there are two possible explanations for this. The first is that these larger networks are *too* deep; the extra layers increase the networks' abilities to memorise and overfit (section 2.9, page 84) to the data they have been trained on, and this dominates any improvement in the networks' processing capabilities. The second explanation is that structures comprising these deeper ResNets (50, 101 and 152) are inherently unsuited towards this task. These networks, as discussed (section 5.3.5.2, page 149 and Figure 5-4), use 'bottleneck' layers with different convolutional layers, with filters of size 1. In the setting of 2D CNNs, these layers are highly efficient providing good performance with a modest increase in the number of parameters per layer. This has allowed increasingly deep networks to be created. However, ResNet 50, which is identical in superstructure to ResNet34, but using bottleneck blocks instead of basic blocks (Figure 5-4) performs *worse* than ResNet34. This is a highly surprising finding, which may have consequences reaching far beyond this example.

Second, adapting the ResNet50 and 101 architectures to the more advanced WideResNet and ResNeXt improved the performance of these networks (Kappa 0.37 for ResNet50, 0.46 for

WideResNet50, and 0.53 for ResNeXt50; Table 5-2), but they still did not reach the performance of ResNet34.

Finally, 1 dimensional adaptation of ResNet34 exceeds the performance of the bespoke 1 dimensional neural network architecture used in Hannun *et al.*'s 2019 *Nature Medicine* paper. This neural network does *not* use bottleneck layers, and is closer in design ResNet34. However, its most notable design difference from the other networks in this paper is the inclusion of dropout layers within every block. Dropout layers are used to reduce overfitting in 2D image processing (section 2.9, page 84), but I found that their introduction significantly hampered the performance of networks in this 1D setting.

5.5.4 Saliency mapping not only highlights salient information, but may help identify errors

The previous chapter of my thesis demonstrated how saliency mapping could have a role in identifying useful features within data (section 4.5.3, page 129). If such features were previously not known to be useful, this may actually further our understanding, and assist with human decision making.

Figure 5-7 also illustrates this point, but in addition it shows how when the neural network *fails* to correctly classify an example, the saliency map appears incongruous. Panels B and D of the figure show two examples of NS-HBP. In panel B, the neural network has correctly classified the beat, and the saliency maps indicate it has used similar features to come to this conclusion as an expert human would. In panel D, however, the neural network has misclassified the beat as MOC, and the saliency map instead shows internal features of the QRS to be particularly salient. Our current understanding of this task cannot explain why such features might be useful to the neural network. This therefore raises the possibility that saliency maps could be reviewed by

humans to allow the neural network's opinion to be 'vetted', and to be treated with a more appropriate level of scepticism if the 'attention' of the network appears to have been drawn to less typical features. I plan on investigating this further with a larger dataset, to see if saliency maps can be processed to independently predict classification errors.

5.5.5 Study limitations

This is a proof of concept study which demonstrates that a neural network can be trained to discriminate between the different ECG responses to His bundle pacing. However, the accuracy of the current system is only 75%. This means the system is not ready for clinical use. However, we believe this system is worthy of ongoing interest for several reasons. First, this current approach is a proof of concept using a small research database of less than 100 cases, and we are planning on expanding this. Second, the current approach simply uses a single QRS complex to make its prediction; meaning (1) it can be used in the pacemaker laboratory without any equipment other than a 12 lead ECG, and (2) this raises the possibility of further performance improvements by introducing more complex inputs to the network, like the patient's baseline ECG. Third, our solution is fast, with single beats classifiable in less than a 10^{th} of a second; real-time ECG analysis is therefore possible.

There was significant class imbalance in our dataset. For this reason, we reported the performance of the classifier in term of Cohen's kappa, which accounts for such an imbalance. However, such imbalances still influence the ability of the neural network to learn, because the network is less likely to predict a rarer ECG class.⁶⁸ I tried to compensate for this by assigning class weights in a 5:1:1 ratio between MOC, NS-HBP and S-HBP, effectively making the neural network's adjustments 5 times stronger for MOC cases, but this increases the risk of the network being able to overfit to the relatively modest number of MOC beats. Expanding our dataset in the future will hopefully, at least partially, alleviate these concerns.

Finally, this was a single centre study using a single expert to label data. This introduces potential biases, both in case characteristics, and the classification of beats. Our centre is conducting a trial for His bundle pacing for heart failure patients, and the relative proportion of patients undergoing the procedure for other indications, such as high degree atrioventricular block, is comparatively modest. Furthermore, even humans using algorithmic approaches to ECG classification have been shown to provide dramatically different classifications,⁹³ and so some “misclassifications” by the network may merely be a consequence of inaccurate classification by an expert in a difficult case.

5.6 Conclusion

In this Chapter I have shown a neural network is able to successfully discriminate between three different ECG responses to His bundle pacing. A 1-dimensional network, which I adapted from a 2-dimensional design, is able to outperform the current state of the art neural network for ECG classification. Finally, I have shown provisional data indicating that saliency mapping might be useful for assessing the certainty with which a neural network’s prediction can be trusted.

6 Echocardiogram view identification

The research covered in this chapter has been previously published as:

Howard JP, Tan J, Shun-Shin MJ *et al.*

“Improving ultrasound video classification: an evaluation of novel deep learning methods in echocardiography.”

J Med Artif Intell. 2020 Mar 25;3. pii: 4. doi: 10.21037/jmai.2019.10.03.⁹⁴

6.1 Abstract

6.1.1 Background

Echocardiography is the commonest medical ultrasound examination, but automated interpretation is challenging and hinges on correct recognition of the ‘view’ (imaging plane and orientation). Current state-of-the-art methods for identifying the view computationally involve 2-dimensional (2D) convolutional neural networks (CNNs), but these merely classify individual frames of a video in isolation, and ignore information describing the movement of structures throughout the cardiac cycle.

Here I explore several novel CNN architectures which are inspired by advances in human action recognition and assess their performance on echocardiogram view identification.

6.1.2 Methods and Results

I extracted 9098 echocardiographic videos across 374 studies. 8732 of these videos (96.0%) were classifiable as 1 of 14 standard designated views. I trained 4 different classes of CNNs to identify the view present in these videos: (1) “classical” 2D CNNs, (2) time-distributed 2D CNNs, (3) 3-dimensional (3D) CNNs, and (4) “two-stream” CNNs processing both spatial and temporal data.

I demonstrate that these new architectures more than halve the error rate of traditional CNNs from 8.1% to 3.9%. I then show evidence that these advances in accuracy may be due to these networks’ abilities to track the movement of specific structures such as heart valves throughout the cardiac cycle. Finally, I show the accuracies of these new state-of-the-art networks are approaching expert agreement (3.6% discordance), with a similar pattern of discordance between views.

6.1.3 Conclusion

Neural network architectures which are able to analyse echocardiographic videos both spatially and temporally, rather than a series of independent frames, can more than halve the error rate for view classification. The novel architectures evaluated here draw on the very latest state-of-the-art methods for video classification in other academic fields. This is the only study which has reported the relative performances of different 2D CNN architectures on a “real world” echocardiogram dataset, which is invaluable data for any group wishing to implement their own view classification system.

6.2 Introduction

Echocardiography is the commonest use of medical ultrasound. It is a key diagnostic test in cardiology, particularly for assessing left ventricular function and valvular disease. Through a combination of portability, availability and affordability, over 10 million echocardiograms are performed each year.¹⁴ However, the limited number of skilled staff available to interpret echocardiograms is a bottleneck.^{15,16}

Efforts are being made to streamline the time-consuming process of reporting and automate the analysis of studies to allow sonographers to scan more patients. One major barrier, however, is that each study comprises upwards of 50 video loops providing depictions of slices of the heart in a variety of anatomical planes and orientations, also known as ‘views’. Before each video loop can be analysed, the view it represents must be correctly identified. One possible approach for this task is using neural networks, which have been employed successfully in several other fields of medical image classification.^{32,41,67}

Neural networks are computer models comprising a large series of simple units known as neurons which are tuned in such a way that their output is useful in decision making.

Convolutional neural networks (CNNs) are a particularly useful form of neural network for computer vision challenges because they have built into their design the notion that adjacent image pixels are meaningfully related to each other in the sense that they build up larger features. Their operation may resemble that of the human visual cortex, which is capable of rapidly learning and performing such tasks.

However, the accuracy of echocardiogram view classification using neural networks has been variable.^{25,26} One recent study using CNNs reported accuracies of only 84%,²⁶ noting particular difficulties in distinguishing between several different views that are taken from the apex of the heart with different probe orientations. Failing to distinguish these views leads to erroneous

identification of the cardiac segments, which are an important basis for assessing myocardial infarction or ischaemia. If further automated analysis is to be done after view classification, it would rely on the view classification system being highly accurate.

Efforts so far have shared a common approach of treating echocardiographic videos as merely a collection of still images. In other words, the CNNs have classified individual video frames in isolation, missing the opportunity to use temporal information of how features such as heart valves or ventricular walls move during the cardiac cycle.

First, I assess the relative performance of several classical CNN architectures. We then introduce some new architectures, inspired by current work in the field of human action recognition, which can process both the spatial and temporal information contained in video (Figure 6-1). The four groups of architectures assessed are as follows:

1. Classical CNN. I assess 5 different CNN architectures, each of which has held the title of being the state-of-the-art in network design for image recognition. As with previous studies, accuracy is calculated using the modal prediction across multiple video frames.²⁵
2. Classical CNN encapsulated within a time-distributed layer. In this design, an entire video is passed through a classical CNN, frame by frame, and CNN's output from each frame is collated. Then each output is fed, sequentially, into a second neural network. The information being transferred from the first CNN to the second is not images, but rather a description of the features present in the image and how they change over time.

3. 3D CNNs. I assess the ability of a three-dimensional (3D) CNN which comprises filters which not only scan ('convolve') through the two dimensions of each frame of a video, but also across frames in the third dimension.
4. "Two-stream" CNNs. Finally, this type of network takes in two streams of data: a spatial stream, and a temporal stream. The spatial stream processes sequential video frames and comprises either a time-distributed classical CNN network (see 2) or 3D CNN (see 3). A second 'temporal' network is also trained, which receives optical flow data, i.e. data describing the movement of objects between frames. The final view decision is based on integration of both temporal and spatial signals.

Finally, I compare the error rate of the best performing network with the disagreement rate of two echocardiography experts, which may represent the upper limits of what is achievable using retrospective human-labelled data.

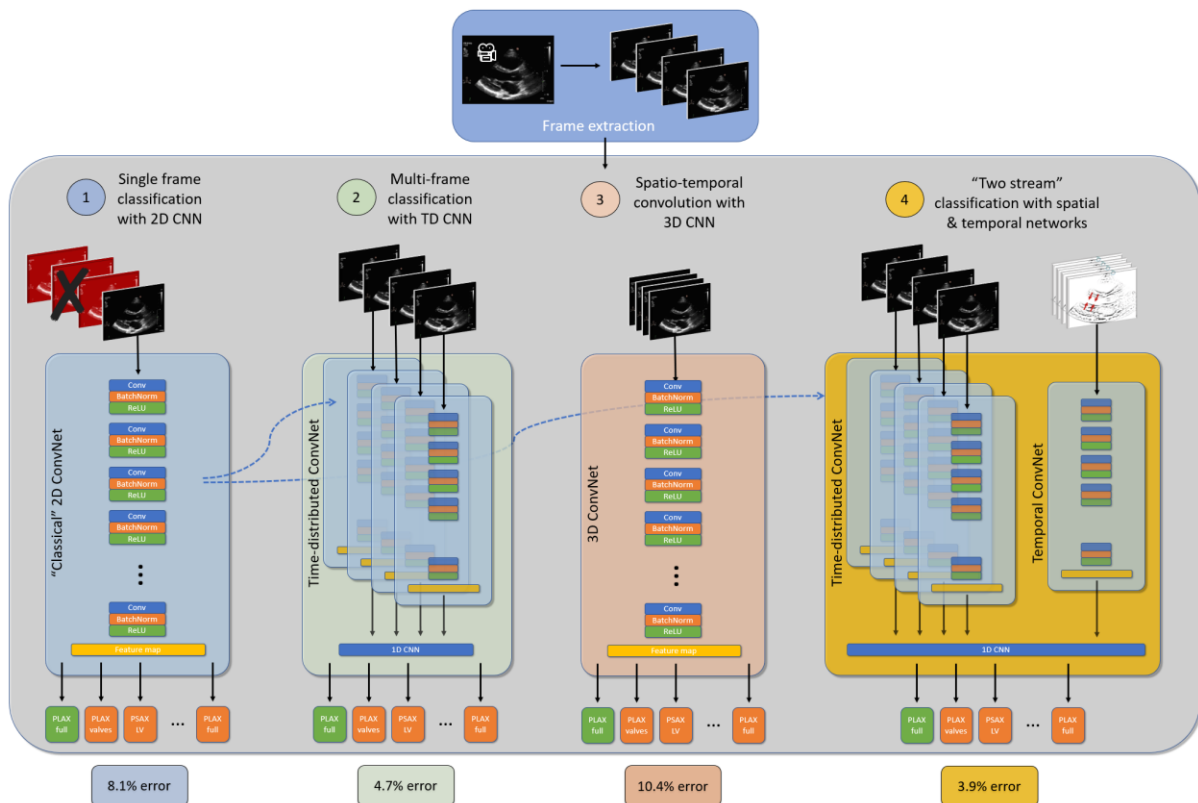


Figure 6-1 - The four different types of neural network architectures used in this study, along with the lowest error rate of each model within each group. The best-performing neural network was a "two-stream" network using both spatial and optical flow inputs, with a corresponding error rate of only 3.9%. Conv = Convolutional layer; Batch Norm = Batch Normalisation layer; ReLU = Rectified linear unit layer. Reproduced with permission from Howard et al. (J Med. Artif Intell, 2019).⁹⁴

6.3 Methods

6.3.1 Data sources

I downloaded a sample of echocardiographic studies from the database at Imperial College NHS Trust. These studies comprised the video loops in the Digital Imaging and Communications in Medicine (DICOM) format and the final scan report in portable document format (PDF).

Exclusion criteria were incomplete demographic information in the report, and the administration of intravenous contrast. All videos which contained at least 40 frames were included in the study.

Studies were randomised in a 3:1 ratio to the training and testing sets. 20% of the training set was used as a ‘validation’ set, against which optimisations in the neural network’s architectures could be assessed during the development stage.

Regulatory approval was gained from the Health Regulatory Agency (Integrated Research Application System identifier 243023).

6.3.2 Data processing

All videos were anonymised prior to processing. This anonymisation was automated and performed by identifying the ultrasound machine model using the DICOM file meta-data. Using this information, I was able to blank the range of pixels that would display the patient’s name, sex and hospital number.

The first 40 frames of each video were extracted and resized to a resolution of either 299 by 299 pixels, or 224 by 224 pixels, depending on the requirements of the network being trained.

Optical flow frames were extracted from videos using denseflow⁹⁵ at a resolution of 224 by 224 pixels. Prior to training, images were normalised so that pixel values ranged between 0 and 1.

6.3.3 Human image classification

I classified each video from the training and testing sets into one of 14 categories which are outlined in Figure 6-2. These classifications (labels) were used to train and assess the performance of the network. Videos felt to show no identifiable echocardiographic features, or which depicted more than one view, were excluded.

I predicted that training the network would require many thousands of example videos. For this reason, I designed software for a labelling pipeline to allow myself to label the view for each video more rapidly. This system involves two steps: (1) hierarchical manual labelling, and (2) computer-assisted labelling.



Figure 6-2 - The 14 echocardiographic views, with example echocardiographs and schematics below. A2CH = apical 2 chamber; A3CH = apical 3 chamber; A4CH = apical 4 chamber; A5CH = apical 5 chamber; Ao = Aorta; AV = aortic valve; IAS = interatrial septum; LA = left atrium; LV = left ventricle; PLAX = parasternal long axis; PS = parasternal; PV = pulmonary valve; RA = right atrium; RV = right ventricle; TV = tricuspid valve. Reproduced with permission from Howard et al. (J Med. Artif Intell, 2019).⁹⁴

6.3.3.1 Step 1 – Hierarchical manual labelling

Manual rapid labelling was facilitated by using a hierarchical approach where each video was classified at two levels. Figure 6-3 shows a simplified example of how this system works, with 6 views rather than the total 14.

At any one time, the reporter (myself) was only be faced with a maximum of 5 valid options. In Figure 6-4, the example case shows a parasternal short axis view, focussed on the pulmonary valve. The user would therefore selection option 2 (parasternal), either with the mouse, or more quickly using the number 2 on the keyboard. The reporter would label a series videos at this level of hierarchy, before moving on to sub-classify all of the classified videos by their specific view. In Figure 6-5 we can see an example of the next step, were a video previously classified as a parasternal short axis is now being further subclassified as depicting the aortic valve (PSAX_AV), the left ventricle (PSAX_LV), or the pulmonary valve (PSAX_PV). This ensures the labelling process requires only simple decision making between a small number of categories at any stage.

6.3.3.2 Step 2 – Computer-assisted labelling

Once I had labelled 500 videos using the manual labelling system in Step 1, I then trained a neural network to attempt to classify the remaining videos in the dataset. With only 500 examples, the accuracy of this system was only around 75%. However, this allowed it to predict a ‘provisional’ label for each remaining video for labelling.

I then wrote a piece of software to rapidly review these labels. It did this by displaying 12 examples of a single specified class on the screen (Figure 6-6). Because only 1 view of video should be depicted on the screen at any one time, this made rapid reviewing a much easier task. I then clicked on any incorrect videos, to mark them as wrongly labelled by the computer, before

clicking 'OK' at the bottom. Any videos which were marked as wrongly labelled would be moved into a pool for hierarchical manual relabelling, as used in Step 1 (page 178), whilst the remaining videos would enter the labelled dataset with the labels provided by the computer. Finally, a further 12 videos would be displayed, and the process would repeat. Initially, as the system was only 75% accurate, each screen of 12 videos would typically contain 3 mistakes.

After reviewing a batch of 500 computer-assisted labels and re-labelling any incorrect predictions, I would then retrain the network. The new network should show improved performance, as it had 500 more examples to learn from than the previous model. I would then re-run the computer-assisted labelling software on the remaining unlabelled videos, and even fewer videos would have been incorrectly labelled by the system.

This approach greatly sped up the labelling process, allowing me to label over one video per second once the network achieved accuracies of 90%.

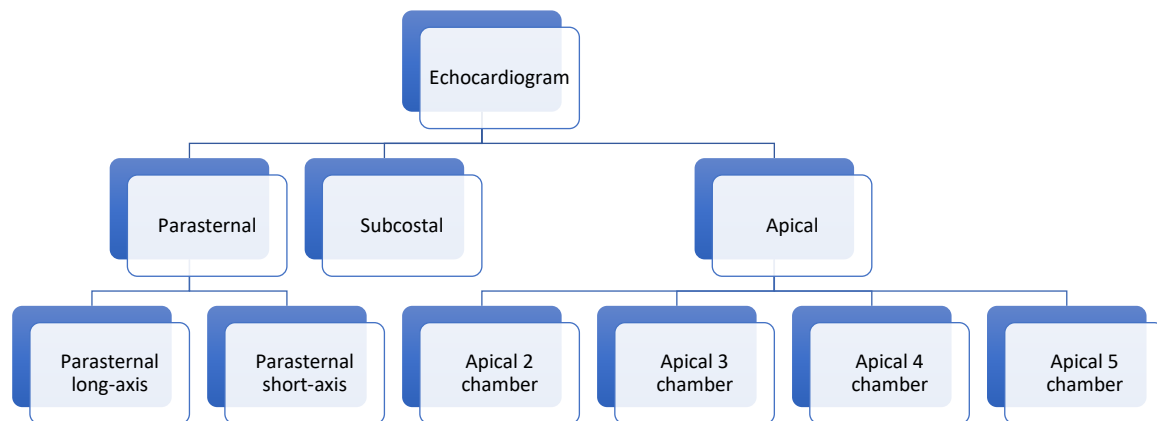


Figure 6-3 - A simplified hierarchical model like that used in the rapid reporting software. Each level of hierarchy represents a stage of labelling for a video. This approach ensures only a maximum of 5 options are presented to the user for any video, facilitating rapid and accurate labelling. In practice, I used a total of 14 views, rather than the 6 shown on this diagram.

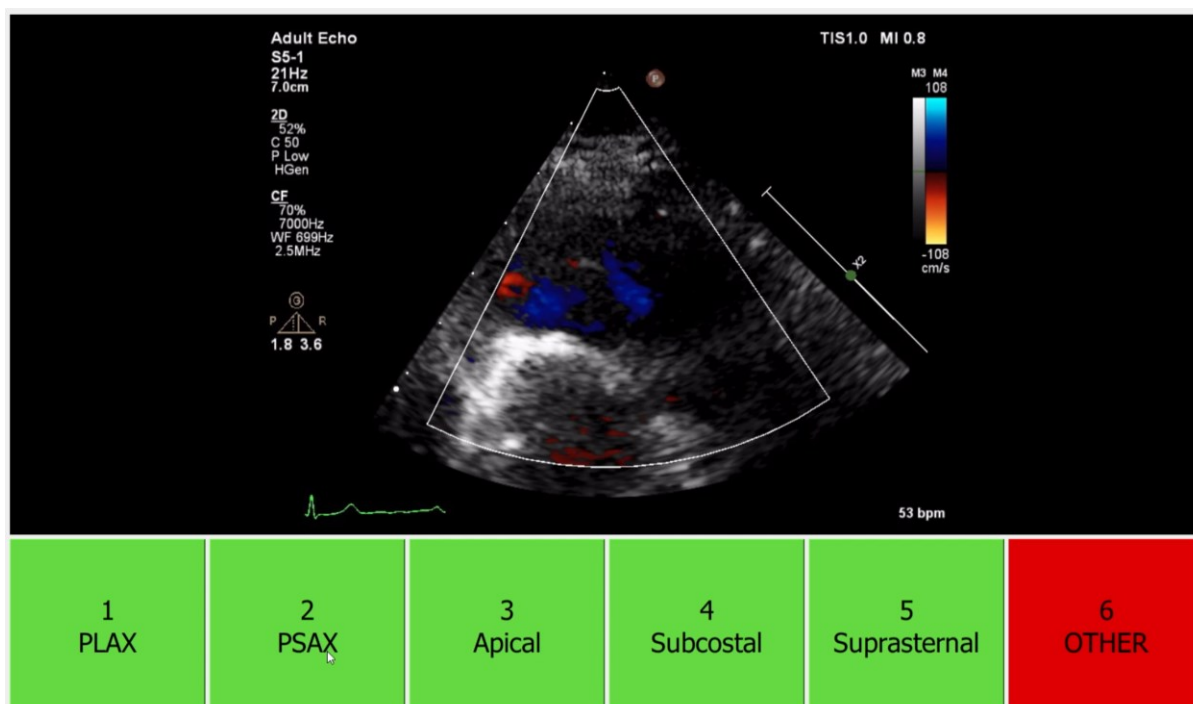


Figure 6-4 - An example of the rapid labelling software I have created showing an example video playing with buttons for the appropriate categories. In this image, a view is being classified using the top level or hierarchy (where the probe is positioned), rather than the precise view. This minimises the number of choices available to the human labeller. Each category can be clicked as a button or selected by pressing the appropriate number (1 to 5 in the example above) on the reviewer's keyboard. The example video is a parasternal short axis video (PSAX) focused on the pulmonary valve, and so the user should either click the 'PSAX' button, or press '2' on their keyboard.

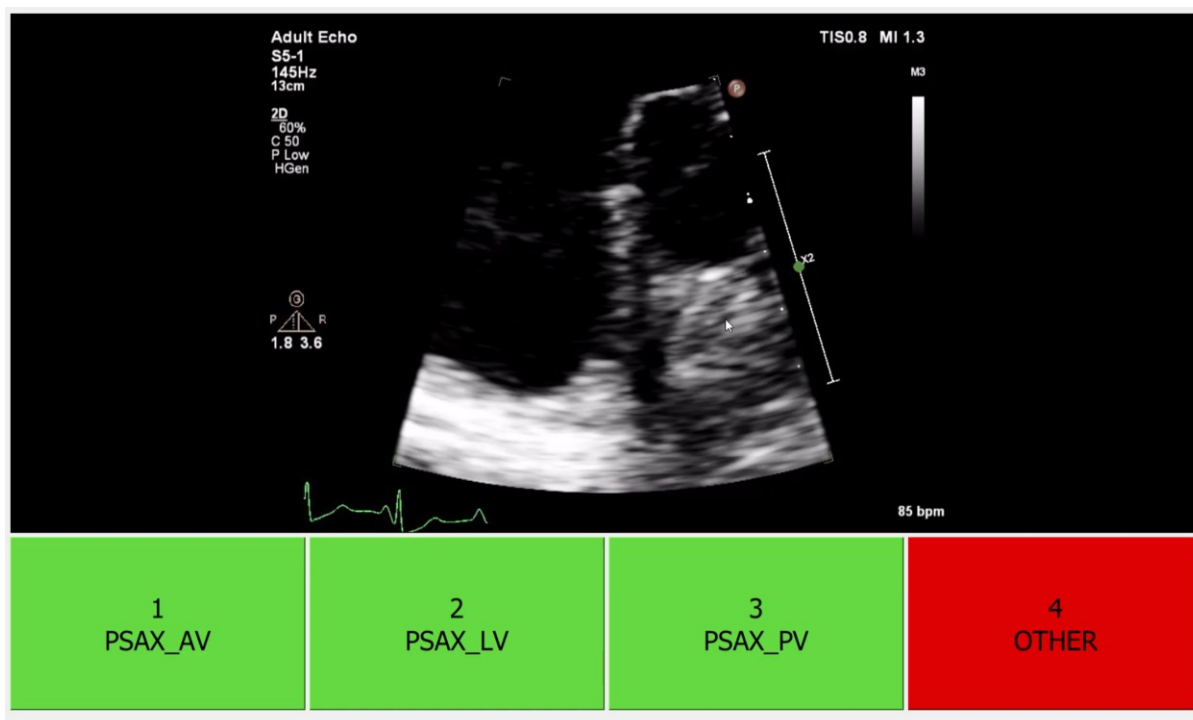


Figure 6-5 – Another example of the rapid labelling software. Here, only parasternal short axis videos (labelled using the previous step) are shown to the human labeller, and videos are now classified according to the exact view depicted. This again limits the number of options from which the human must choose, and greatly simplifies the task. In this example, the video shows the view is focused on the aortic valve (AV), and so the user should either click the ‘PSAX_AV’ button, or press ‘1’ on their keyboard.

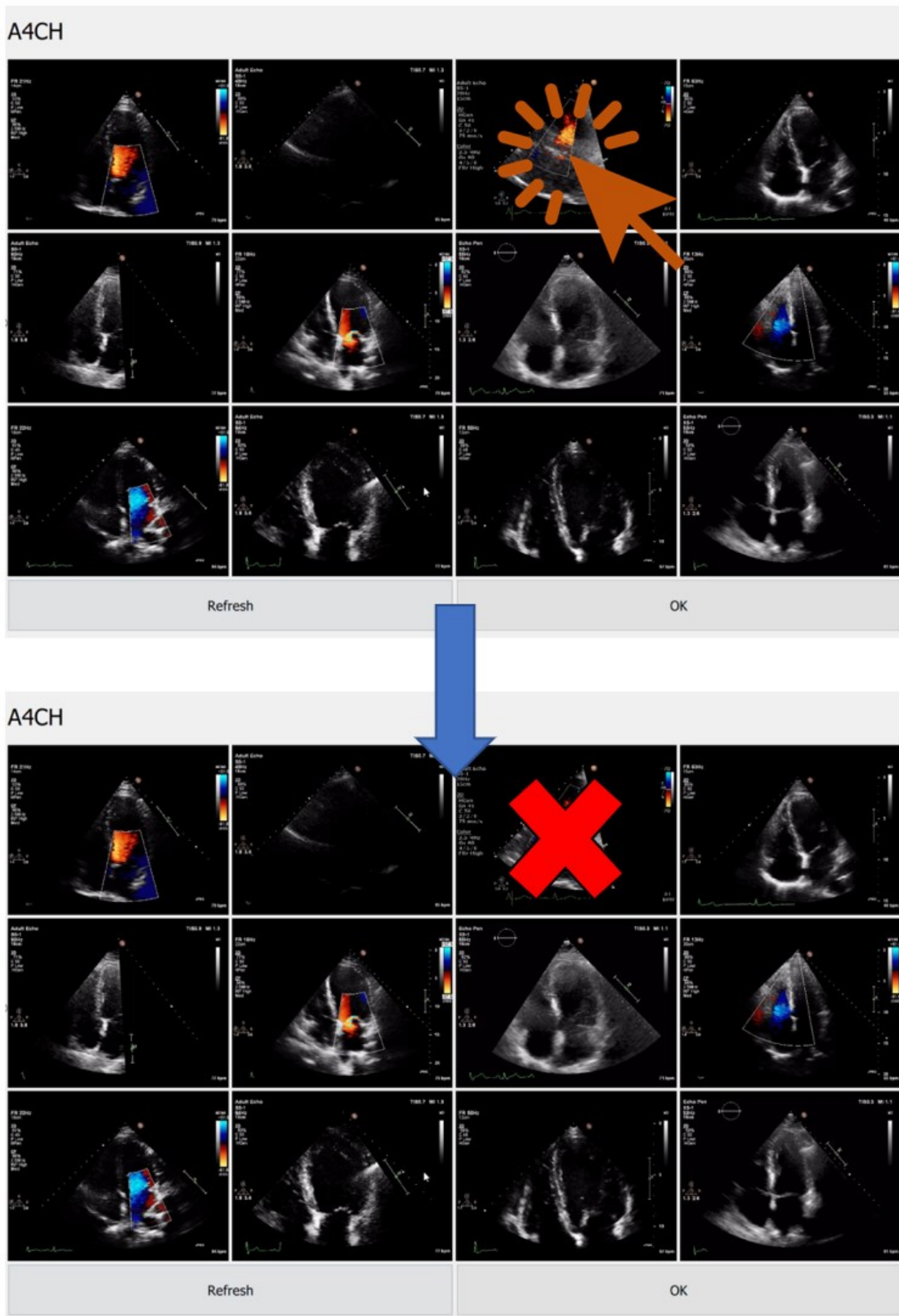


Figure 6-6 - Computer-assisted labelling. In this example, the computer has found 12 videos which it believed are Apical 4 Chamber videos (A4CH). The reporter reviews these, and notes a single mistake; the third video on the top row is actually

*an Apical **5** Chamber. The reporter clicks on this and marks the video for manual relabelling at a later date. The remaining 11 videos enter the dataset with the label ‘Apical 4 Chamber’.*

6.3.3.3 Reporting by a second physician - reproducibility of labels

A second physician, a consultant cardiologist, classified each video from the test set, unaware of the other human’s classifications, so that inter-expert agreement could be assessed. This dual reporting of the test set enables several analyses. First, the agreement of a neural network with the first expert’s opinions represents the ability of the network to reproduce the decision-making process of the human who trained the network and may represent the best measure of a specific neural network’s inherent discriminant ability. Second, the agreement of a neural network with the second expert’s opinions may provide an estimate of the “external” accuracy of the network, indicating how it performs when judged against a cardiologist who did not contribute towards its training. Finally, the rate of disagreement between the two experts represents the proportion of cases where a consensus could not be reached and may represent an upper limit on how well a ‘perfect’ neural network, which agrees completely with the human that trained it, can agree with another expert.

The second physician reported all videos in the testing set using the computer-assisted labelling system, before using hierarchical manual labelling to re-assign any incorrect labels.

6.3.4 Neural network details

6.3.4.1 Classical 2D CNN

For the classical 2D CNNs, we investigated the performance of 5 different neural network architectures: DenseNet, Inception V3, ResNet, VGGNet and Xception. Each network was initialised using weights derived from training on ImageNet, a large image database used for object recognition. The final convolutional layer of each network was fed into a global average pooling layer before the final output layer. During testing, 5 linearly spaced frames were chosen from each video and the modal answer was used when calculating accuracy. The best performing

architecture (Xception) from this stage was used as the spatial convolutional base for the time-distributed and two stream networks.

6.3.4.2 Time-distributed 2D CNN

This neural network contains the 2D CNN of the Xception architecture which we trained in the previous stage. Into this ‘inner’ network I fed the first 20 frames of each echocardiogram video. For each frame, I take the results of the penultimate layer of the network (so not the network’s actual prediction, but rather the ‘feature map’ going into the final prediction layer), and create a sequence from them. This sequence of 20 feature maps is then fed into a 1-dimensional convolutional neural network, which I predicted would be able to learn to identify the view present not only from which features are present in each frame, but how the features change throughout the cardiac cycle.

More specifically, I extracted the feature maps from the global average pooling layer for each frame, and these were fed into three 1-dimensional depthwise-separable convolutional layers comprising 256 kernels of filter size 3, separated by 1-dimensional max pooling layers of pooling size 2. Finally, a 1-dimensional global average pooling layer was used to connect the final convolutional layer to a densely connected layer for classification.

This design was inspired by the ‘long-term recurrent convolutional network’ design pioneered by Donahue *et al.*,⁹⁶ but I have made several changes. First, I used a series of depthwise-separable convolutional layers instead of long-short term memory layers. Second, I used a 1-dimensional global average pooling layer with dropout to connect the classification layer to the convolutional layers. I found that these changes greatly reduced the number of parameters in the model, which I hypothesised would reduce the tendency of the network to overfit.

6.3.4.3 3D CNNs

Two 3D CNN architectures were assessed. The first 3D CNN was a modified form of the C3D architecture pioneered by Tran *et al.*³³ Given this architecture would require over 1 billion trainable parameters when passed images of the dimensions used in this study, I swapped the flattening layer for a 3D global average pooling layer which has since become the modern practice⁹⁷ and which also drastically reduces the number of parameters. The second 3D CNN was created using the Inception 3D (I3D) architecture as published by Carreira *et al.*⁹⁸ The network was pretrained on ImageNet and Kinetics using weights released by Deepmind.⁹⁹ Model weights were frozen for the first epoch before unfreezing weights progressively backwards over 5 epochs to minimise catastrophic forgetting of learned weights during early gradient updates. Both networks received the first 20 frames of each echocardiogram.

6.3.4.4 Two-stream networks

Finally, this two-stream design comprises two separate convolutional neural networks, or ‘streams’ (one ‘spatial’ stream, one ‘temporal’ stream), which process a video’s spatial and temporal features separately. The two streams are then joined, allowing the spatial and temporal data to be integrated when coming to the final decision for what view is present.

The spatial stream comprised either a time-distributed 2D CNN or a 3D CNN, trained during the previous steps. The spatial stream was fed the first 20 frames of each echocardiogram, as a video.

The temporal stream comprised a separate classical 2D CNN architecture which was trained to identify videos using only optical flow data. The optical flow data were actually merely greyscale images, where the brightness of the image corresponds to how much the structures at that location have moved, compared with frame before it. A single data sample fed into the temporal

network was made up of data describing the inter-frame optical flow from a series of 10 frames. I actually exported optical flow images separately for movements in the x (horizontal) and y (vertical) dimensions. This resulted in a 20-channel input image of 224 by 224 pixels for a 10 frame ‘chunk’.

I then encapsulated this temporal 2D CNN in a time-distributed layer, using the same method that I employed for the time-distributed 2D CNN above. This allowed me to feed in 4 sets of 10 frame chunks from a video into the network.

Finally, the spatial and temporal streams were joined together. This was done by concatenating the outputs of each of the streams’ global average pooling layers, and then joining these to a densely connected output layer of 14 neurons.

This design is inspired by the two-stream networks pioneered by Feichtenhofer *et al.*¹⁰⁰, I made several changes. Most notably, the network comprising the temporal stream was an untrained Inception V3⁷⁰ model. Furthermore, unlike Feichtenhofer’s implementation, this network contains no fully connected layers, as I wished to minimise the risk of the network overfitting. I found this also led to much faster convergence with improved accuracy.

Each network was trained until the validation loss plateaued. Models were saved after each epoch, and the model with the highest validation accuracy was used for the final assessment on the test set. The final output layer of every network comprised 14 densely-connected neurons (one for each view). Loss was calculated using the categorical cross entropy loss function and weights were updated using the Adam optimizer. The batch size for all networks was 20.

Saliency maps were used to further investigate the relative focus of the two components of the two-stream networks (spatial features and temporal features). To allow pictorial visualisation of this, despite the networks receiving input in video format, the saliency patterns across each

spatial frame were averaged and then normalised, before being super-imposed upon a single frame from the video.

Programming was performed with the Python programming language, with the Tensorflow⁶² and Keras⁶³ machine learning frameworks and the Keras-vis package⁷⁵.

6.3.5 Statistical analysis

The primary statistical endpoint was accuracy, defined as the proportion of videos correctly classified according to their view. Confidence intervals for accuracy were calculated using the “exact” binomial method. Significance testing between models was by McNemar’s test with $p = 0.05$ as the threshold for statistical significance, with an exact test used for contingency tables including any counts below 25. Cohen’s kappa was also calculated for each model to account for imbalanced class sizes. Statistical analysis was performed using the R programming language.

6.4 Results

6.4.1 Dataset

I extracted 374 full echocardiographic studies which met the inclusion criteria. 282 videos were randomly assigned to the training set, and 92 videos were randomly assigned to the testing set. These studies included 9098 echocardiographic videos with at least 40 frames. Of these, I was able to classify 8732 (96.0%) videos as one of the 14 designated views (Figure 6-2), with the remaining 366 videos being either completely unrecognisable or the view in these videos changing during the video loop. The characteristics of the studies and the videos they include are shown in Table 6-1.

	Training set	Testing set
Study characteristics	n = 282	n = 92
Sex (male)	144 (51.1)	46 (50.0)
Age	61.7 (18.6)	61.8 (18.5)
Manufacturer		
Philips	224 (79.4)	72 (78.3)
GE	58 (20.6)	20 (21.7)
Video characteristics	n = 6592	n = 2140
Colour flow Doppler	718 (10.9)	234 (10.9)
Classes		
Parasternal long axis	713 (10.8)	260 (12.2)
Parasternal long axis valves	426 (6.5)	123 (5.7)
Parasternal long axis RV inflow	207 (3.1)	63 (2.9)
Parasternal short axis LV	1045 (15.9)	343 (16.0)
Parasternal aortic & pulmonary	874 (13.3)	291 (13.6)
Apical 2 chamber	534 (8.1)	152 (7.1)
Apical 3 chamber	503 (7.6)	145 (6.8)
Apical 4 chamber LV	829 (12.6)	289 (13.5)
Apical 4 chamber RV	222 (3.4)	79 (3.7)
Apical 5 chamber	295 (4.5)	84 (3.9)
Apical LA/MV focused	199 (3.0)	58 (2.7)
Apex	25 (0.4)	9 (0.4)
Subcostal	570 (8.6)	193 (9.0)
Suprasternal	150 (2.3)	51 (2.3)
Unclassifiable *	273 (NA)	93 (NA)

*Table 6-1 - Baseline characteristics of study-level and video-level parameters. Categorical data is shown as numbers (%); continuous data are shown as mean (standard deviation). * 273 and 93 videos from the training and testing sets, respectively, were unable to be classified, and do not contribute towards the class percentages; this is because they showed two or more views during the full video loop, or neither of the human operators were able to identify sufficient landmarks to classify the video.*

6.4.2 The performance of the different neural network architectures

5 different classical 2D CNNs were compared. The best performing architecture was Xception, with an error rate of 8.1% (Cohen's Kappa 0.910). For this network, the two distinctions that were most error-prone were (a) 20 cases of confusing A4CH with A5CH and (b) 19 cases of confusing PLAX inflow with parasternal aortic and pulmonary valves. The Inception V3, ResNet and DenseNet 121 architectures performed with similar accuracy. The network based on the VGGNet architecture, however, failed to develop into a meaningful classifier. The accuracies and the total number of trainable parameters for each network are shown in Table 6-2.

The time-distributed CNN performed better than the best performing 2D CNN ($p < 0.0001$) with an error rate of 4.7% (Cohen's Kappa 0.947).

The two 3D CNNs examined gave disparate results. The modified C3D network performed poorly (Table 6-2) and failed to provide any meaningful classification. However, the I3D network yielded an accuracy superior to that of the 2D CNN on which it was based (Inception 3D's 10.6% error versus Inception 2D's 11.9% error). However, this network proved inferior to time-distributed CNNs ($p < 0.0001$).

The two-stream networks demonstrated the highest accuracies, with the network based on the time-distributed CNN having an error rate of only 3.9% (Cohen's Kappa 0.957). This performance was superior to that of both the classical 2D CNNs and 3D CNNs ($p < 0.0001$), though superiority versus the time-distributed CNN did not reach statistical significance ($p = 0.053$). For this network, the two distinctions that remained most error-prone were (a) 10 cases of confusing A4CH with A5CH and (b) 8 cases of confusing A2CH with A3CH.

Confusion matrices for each class of network are shown in Figure 6-7 (panels A to D). The changes in classifications associated with using the best performing model, rather than a classical 2D CNN are shown in panel E.

Architecture	Trainable parameters	% Error	Cohen's kappa
Classical CNNs			
VVGNet 16	14.7 million	86.4	0
Inception V3	21.9 million	11.9	0.868
Resnet	23.6 million	11.0	0.878
DenseNet 121	7.0 million	10.0	0.889
Xception	20.9 million	8.1	0.910
Time-distributed CNN			
TD Xception	21.5 million	4.7	0.947
3D CNN			
C3D	46.6 million	0	0
Inception3D	12.3 million	10.6	0.882
Two-stream networks			
Temporal stream (optical flow)*	21.8 million	34.8	0.622
Two-stream (Inception3D)	34.1 million	9.6	0.901
Two-stream (TD Xception)	42.7 million	3.9	0.957

Table 6-2 - Table demonstrating the total number of trainable parameters, accuracy on the test set, and associated Cohen's kappa for each of the networks on the test set. * = This figures reflect the accuracy of the temporal network alone, which goes on to form one stream of each of the two-stream networks assessed. TD = Time-distributed.

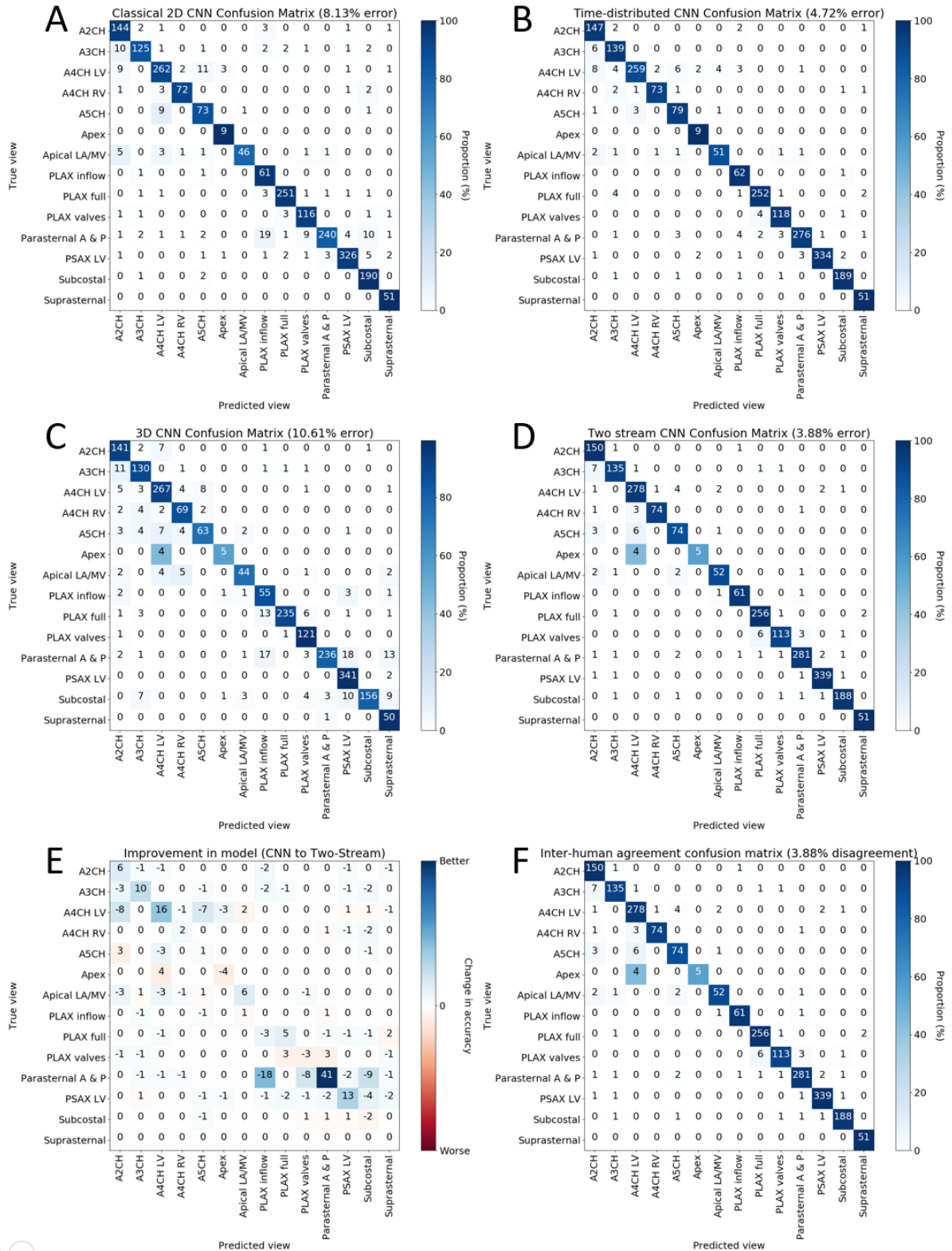


Figure 6-7 - Confusion matrices for the best-performing classical CNN model (A), time-distributed model (B), 3D CNN (C) and two-stream network (D). The improvement associated with using the two-stream network versus the classical CNN is shown in panel E. The inter-human agreement confusion matrix is shown in panel F.

6.4.3 Inter-cardiologist agreement

The results in Figure 6-7 show disagreement (or error) rates of only a few percent, and these errors appear predominantly clustered between certain pairs of views which represent anatomically adjacent imaging planes. It was conceivable, therefore, that some of the residual apparent error was due to an inherent difficulty of deciding between views that are similar in appearance and are in spatial continuity.

To investigate this, I compared the test set classifications of the second expert with those of the first expert.

Of the 2140 test set videos, 74 (3.6%) were classified differently by the second expert (Figure 6-7; panel F). The two distinctions that caused the most disagreement were (a) 13 cases of A5CH versus A4CH and (b) 10 cases of A3CH versus A2CH.

When allowing either expert classification as correct, the error rate of the neural network decreased to only 2.6%.

Finally, addressing only the 2064 test videos classified identically by the two cardiologists, the error rate of the two-stream network was only 2.2%, with only 45 videos “misclassified” by the network.

For reference, the two-stream network’s error rate when judged solely against the second human’s classification was 4.3%.

6.5 Discussion

This study, the most comprehensive of echocardiographic view classification to date, shows that the application of new CNN architectures can reduce the error rate of echocardiographic video classification by more than two-fold. It further suggests the remaining error rate may contain a substantial element of “judgement calls”, where experts are uncertain and when forced to commit to a class independently, pick a different class from other experts.

6.5.1 Temporal neural networks achieve half the error rate of classical 2D CNNs

I had success with two approaches of integrating temporal information: time-distributed networks and two-stream networks. The two-stream network had an error rate less than half that of the best classical 2D CNN.

Much of this benefit appears to be through improved discrimination between certain pairs of views that classical CNNs find challenging. For example, 19 videos were misclassified by the 2D CNN, but only 2 by the two-stream network, when deciding between the ‘PLAX inflow’ and ‘parasternal aortic and pulmonary valves’ views. On a single frame (Figure 6-8; panels A & B) these videos are easily confused, even by a human expert. One clue to their correct classification is the direction of motion of the leaflets when the valve opens: upwards versus downwards. In most static images it is almost impossible to distinguish, and the true identity only emerges when a temporal sequence of images is examined. The resulting error rate for this ‘parasternal aortic and pulmonary valves’ view fell from 17.5% for the classical 2D CNN to just 3.4% for the two-stream network.

Figure 6-8 panels C & D show saliency maps derived from the two-stream network for this parasternal pulmonary valve example. Saliency maps highlight the features of the videos which are contributing most towards the neural network's decision. Panel C appears to indicate the spatial arm of the network's decision is largely influenced by the anatomical borders of the major structures such as the pulmonary artery and left ventricle (highlighted in cyan). The temporal (optical flow) arm's saliency map (panel D), in contrast, shows intense focal activation over the pulmonary valve leaflets. These visualisations may support the theory that the two-stream network's ability to discriminate between such classes may in part be due to their ability to track the movement of structures such as valves throughout the cardiac cycle.

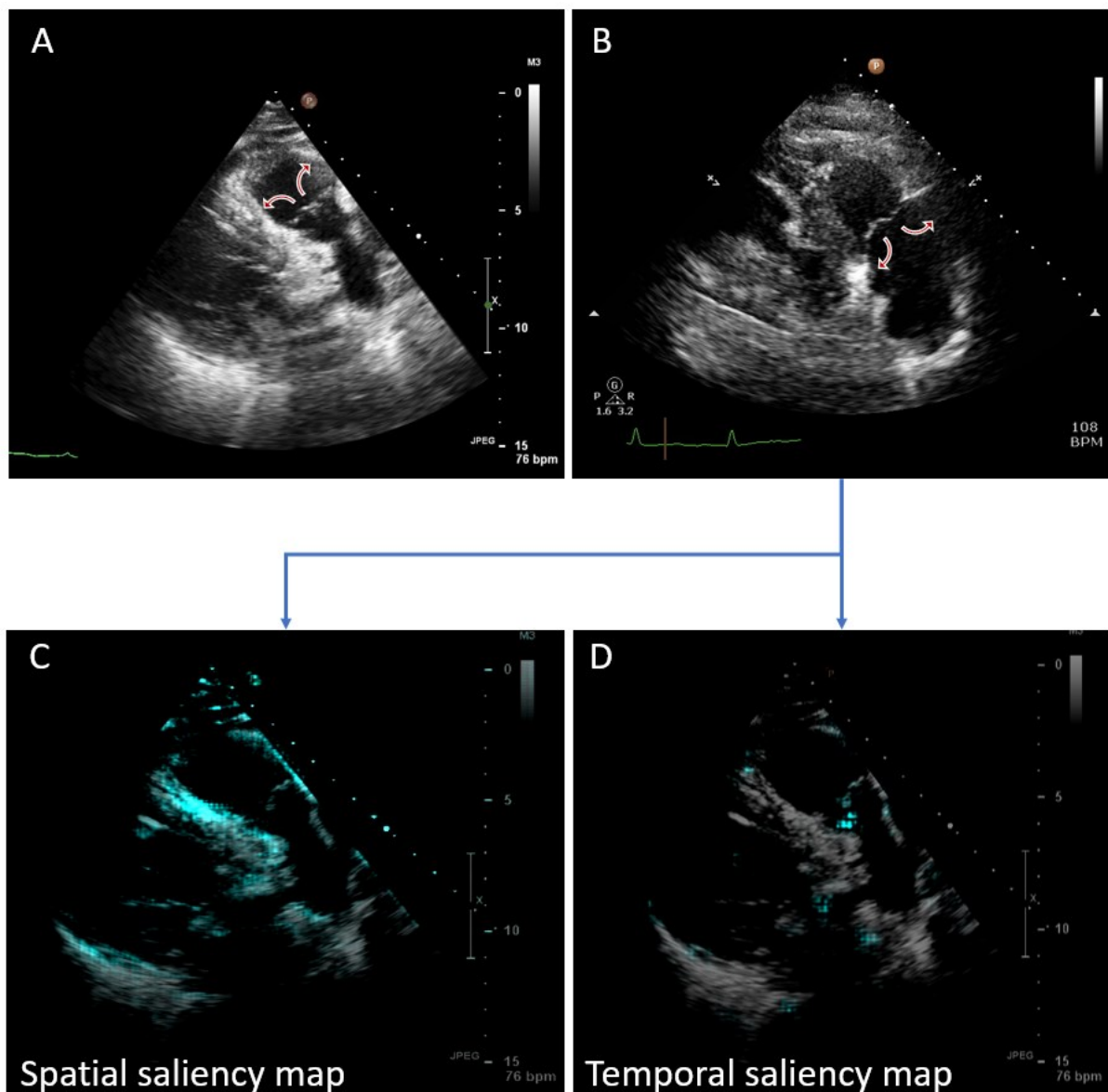


Figure 6-8 - A comparison of a 'PLAX inflow' view of the tricuspid valve (A) and a 'parasternal [aortic and] pulmonary valves' (B) view of the pulmonary valve (also termed 'RV outflow'). These views are almost indistinguishable in a still image. However, when viewed as a video, the echocardiograph on the left clearly demonstrates the valve opening upwards (inwards; see arrows), allowing blood flow into the heart through the tricuspid valve, whereas the echocardiogram on the right shows the valve leaflets opening downwards (outwards; see arrows) allowing flow out of the heart. Misclassifications of these classes were common using classical 2D CNNs, but are almost eliminated by employing temporal models such as the 'two stream' networks. Saliency mapping can be used to visualise how the features from the pulmonary valve video contribute towards the two-stream network's decision. In panels C and D, the important features leading to the classification are highlighted in cyan. Panel C shows the spatial arm of the network appears to use the anatomical borders of the major cardiac structures

present (pulmonary artery and left ventricle). Panel D, however, shows the decision of the temporal arm of the network is overwhelmingly influenced by the optical flow data of the valve itself.

6.5.2 Temporal neural networks show both a magnitude and pattern of errors similar to a human cardiologist

The time-distributed 2D CNN and two-stream networks, which process each 2D image in a video sequentially, avoid the enormous resource consumption of 3D CNNs, which may be why they achieve such high accuracies.

Using one human expert as the reference, the best-performing neural network agreed in 96.1% of cases (3.9% error), while the second expert agreed in 96.4% (3.6% error). If the dataset is restricted to only those images where the two humans agreed, the network was 97.8% accurate, with only 45 (2.2%) video loops classified differently by the network.

Interestingly, the two views the network found most difficult to correctly categorise (Figure 6-7, bottom left) were the also the two views on which the two experts disagreed most often: A4CH versus A5CH, and A2CH versus A3CH. 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.

6.5.3 3D CNNs fail to surpass the performance of classical CNNs

In this study 3D CNNs showed some success in echocardiographic view identification but failed to reach the accuracies reported by even the classical CNNs, which have no ability to process data temporally.

One possible explanation would be that the 3D CNNs were trained on 6592 videos, which is superficially a smaller number than the $\sim 260,000$ frames on which the 2D CNNs were trained. However, the latter figure is merely the result of each video containing 40 frames: the total pixel information is identical. The ratio of data to the number of trainable parameters is therefore of a similar magnitude between the 3D and 2D CNNs.

A more likely explanation is the relative complexity of the classical CNNs and the 3D CNNs. The classical CNNs work by sliding, or ‘convolving’, small two-dimensional templates (termed kernels) through an image, such as a single frame within a video, to find matching regions. A 3D CNN, however, uses three-dimensional kernels which are convolved not only through the two-spatial dimensions of image, but also through time, by sliding them through sequential frames. This allows a 3D CNN to learn not only shapes within a frame of a video, but how these shapes change *through* the video. Unfortunately, this requires vastly more processing power: a classical CNN kernel examining a 7-by-7 pixel area comprises a grid of 7^2 (49) values. An equivalent kernel in a 3D CNN, however, would examine a 7-by-7 pixel area over 7 frames, and would comprise a grid of 7^3 (343) values, greatly increasing the memory, processing and data requirements of training. For this reason, the 3D CNNs in current use are typically ‘scaled down’ versions of their classical 2D counterparts,⁹⁸ with early layers of the network rapidly reducing the resolution of the video through “down-sampling” to reduce the number of pixels the network has to process. The I3D network, for example, contains 171 convolutional layers, but after only the first 3 of these, the data is down-sampled from 299-by-299 pixels to 75-by-75. It may well be this dimensionality reduction that prevents the 3D CNNs (and the two-stream networks incorporating them) from matching the performance of the other networks examined, as they lose the ability to examine fine features early in their processing pipelines.

6.5.4 Utility of accurate view identification

Accurate automation of view identification is necessary for the automated analysis of echocardiograms. These neural networks are therefore likely to form the foundation of future workflow processes for echocardiogram interpretation. However, these networks may also show further utility in echocardiogram acquisition; a neural network, capable of accurately identifying the echocardiogram view, could be used to assist inexperienced sonographers to acquire better views by warning when their acquisition of an apical 4 chamber, for example, appears more like an apical 5 chamber.

6.5.5 Study limitations

Interpreting the results of a neural network study alongside previous studies can prove difficult. There have been two previously published papers assessing the role of classical 2D CNNs for view classification, and they have published very different results, with accuracies of 84%²⁶ and 97.8%²⁵, respectively. There are several possible explanations for the wide range of reported accuracies.

First, the more numerous the view categories, the more difficult the task of the neural network, since if a group of videos are considered a single view in one study but multiple views in another, those multiple views are likely to be relatively similar in appearance. Moreover, one of the studies grouped all images possessing colour as a single category, regardless of the anatomical plane.²⁶ Since identifying the presence of colour is simple, this increases the reported accuracy of any network.

Second, it is possible that previous studies included videos that were more easily differentiated as classical planes, which could explain complete lack of confusion in one study²⁵ between A2CH and A3CH, which are anatomically continuous.

Third, studies sometimes show networks performing better than humans, but have trained and tested the network on reduced-resolution images such as 80-by-60 pixels and have accordingly tested the humans on such low-resolution images, which they will not have had experience of distinguishing.²⁵

For these reasons, I re-implemented 5 different classic 2D CNN architectures, and it is against these which the novel architectures have been assessed. Furthermore, the human experts were provided with full-resolution ultrasound videos, even though the networks were using reduced-resolution data.

6.6 Conclusion

In this study of over 8000 echocardiographic videos, I have shown that switching to advanced neural network architectures can halve the error rate for view classification, almost reaching the concordance achieved by second opinions from blinded human experts. Moreover, the types of misclassification these advanced networks now make are very similar to the sources of differences of opinion between human experts.

7 Automated analysis of cardiovascular magnetic resonance anatomy images

The research covered in this chapter has been previously published as:

Howard JP, Zaman S, Ragavan A *et al.*

“Automated analysis and detection of abnormalities in transaxial anatomical cardiovascular magnetic resonance images: a proof of concept study with potential to optimize image acquisition”

Int J Cardiovasc Imaging. 2020 Oct 29. doi: 10.1007/s10554-020-02050-w.¹⁰¹

An abstract of this work was presented at The Society for Cardiovascular Magnetic Resonance Scientific Sessions 2020, and I was awarded a prestigious and competitive travel scholarship for the work.

7.1 Abstract

7.1.1 Background

The large number of available MRI sequences means patients cannot realistically undergo them all. Therefore, the sequences to be acquired during a scan are planned beforehand based on clinical details. Adapting this to unexpected findings requires an operator who can recognize their significance during the scan, and an automated system could therefore improve the acquisition process.

I investigated whether deep learning of the sequences acquired in the first minutes of a scan could provide an early alert of abnormal features.

7.1.2 Methods

Anatomy sequences from 375 CMR scans were used as a training set. From these, I annotated 1500 individual slices and used these to train a convolutional neural network to perform automatic segmentation of the cardiac chambers and great vessels and to identify pleural effusions.

200 scans were used as a testing set. The system segmented each slice using the network and assembled a 3D model of the thorax, from which it made clinical measurements.

The gold standard was the clinical measurements reported by the clinician using the full scan.

7.1.3 Results

The system was successful in segmenting the anatomy slices (Dice 0.910) and identified multiple features which may guide further image acquisition. Diagnostic accuracy was 90.5% and 85.5%

for left and right ventricular dilatation, 85% for left ventricular hypertrophy and 94.4% for ascending aorta dilatation. The area under ROC curve for diagnosing pleural effusions was 0.91.

7.1.4 Conclusions

Here I present a neural network that can segment a usable 3D model of the thorax from anatomy images acquired in the first few minutes of a scan. It can identify multiple abnormal features such as left ventricular dilatation and hypertrophy, right ventricular dilatation, ascending aortic dilatation and pleural effusions. This early information could improve the patient experience and reduce resource wastage by shortening scan times, facilitating protocol adaptations and prioritizing cases for medical supervision and reporting.

7.2 Introduction

CMR offers limitless scan planes and a large range of different sequences, but this means that the acquisition process must be selective because not every patient can have every imaging plane scanned with every sequence. Efficient clinical practice therefore operates through standardized “protocols”, which list the sequences that are most likely to generate images which confirm or refute important diagnoses for particular clinical indications.²⁸

However, sometimes the scan itself reveals unexpected findings which may (a) indicate that some previously protocolled images are no longer required or (b) indicate that further images should be acquired to confirm or refute a previously unsuspected diagnosis. Some sequences are only interpretable before gadolinium contrast has been given meaning that the opportunity to use them may be lost if the requirement for them is not spotted early.

In recent years, deep learning using neural networks has shown increasing performance in the classification and segmentation of medical imaging data.⁶⁷ Recent work has shown similar precision to human techniques at tasks such as left ventricular (LV) segmentation.^{102–104} These approaches, however, have focused on the segmentation of high-quality LV-dedicated sequences with ideal scan plane orientation.

Within the first couple of minutes of a CMR scan, a series of trans-axial images are routinely acquired, commonly termed the “anatomy” sequences. In this study I investigate whether deep learning methods could analyse these early images to extract information that could (a) help a radiographer recognize the need to modify the ongoing protocol, (b) help identify cases that should be prioritized for medical supervision or early reporting.

7.3 Methods

7.3.1 Data extraction

575 sequential CMR scans from 2 manufacturers (Philips Healthcare, Amsterdam, Netherland & Siemens, Munich, Germany) performed across 2 London hospitals were extracted. Inclusion criteria were adult scans in which a bright-blood axial anatomical sequence had been performed and a valid final report which included the following BSA-indexed (suffix i) measurements as a minimum: LV end-diastolic volume (LVEDV $_i$), LV mass (LVM $_i$) and RV end-diastolic volume (RVEDV $_i$). I did not exclude scans with artefact, e.g. due to atrial fibrillation or breath-holding difficulties as long as the final scan report included the above minimum measurements. We also extracted the following measures from the report where present: the diameter of the ascending aorta; the presence of hypertrophic cardiomyopathy (HCM), the presence of dilated cardiomyopathy (DCM) and the presence of pleural effusions.

The CMR scans were then randomly assigned to different datasets, each serving a specific purpose (Figure 7-1). 200 scans were assigned to the “testing dataset”. This dataset was used to report the final accuracy of the system and was not shown to the neural network at any stage of its training. The remaining 375 scans were assigned to the “training & validation dataset”. This dataset was used to train the neural network. 75% of these scans were used to directly train the neural network (the “training dataset”), and 25% were used to continually appraise the performance of the network during development (the “validation dataset”).

Ethical approval was gained from the Health Regulatory Agency (Integrated Research Application System identifier 243023).

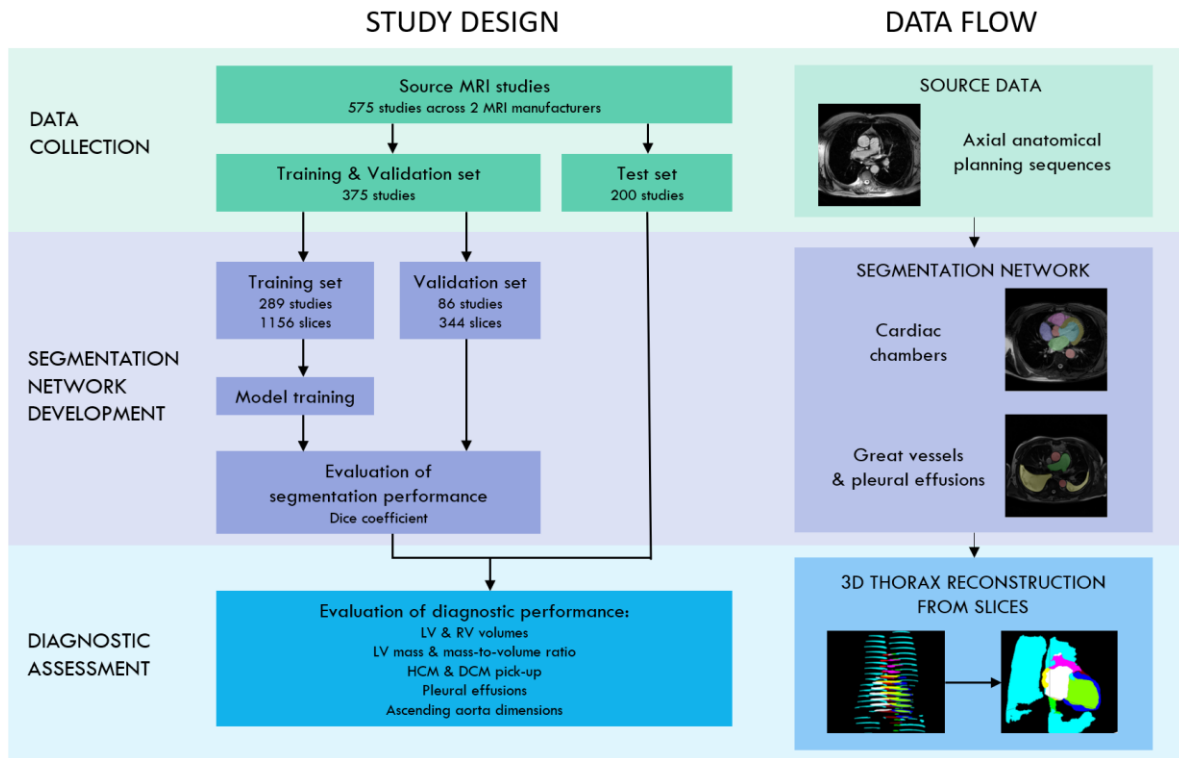


Figure 7-1 - Study design flow chart & data flow. The study comprised 575 studies across 2 manufacturers, which were split into a training and testing set. The training set was used to train an AI which could segment each slice of the anatomy sequence. Then, each scan in the testing set was analysed to allow creation of a 3D model of a patient's thorax. This 3D model was then analysed. 3D = 3-dimensional. DCM = dilated cardiomyopathy. HCM = hypertrophic cardiomyopathy.

LV = left ventricle, RV = right ventricle.

7.3.2 Data processing and labelling

The axial anatomical images were isolated from each scan. The acquisition parameters for these sequences are shown in Table 7-1. 1500 slices were then chosen at random, across the training and validation sets and were then labelled by clinicians using custom-designed software to draw around the following anatomical features, if present: ascending aorta, aortic arch, descending aorta, left atrium, left ventricle, pericardial effusion, pleural effusion, right atrium and right ventricle. Each slice was then resampled to 560 by 560 pixels, with zero-padding for non-square acquired images.

	Siemens	Philips
Field Strength	1.5	1.5
Breath held	Yes	Yes
Flip angle	80	60
Repetition time	357.05	3.109
Echo time	1.39	1.555
Imaging frequency	63.67	63.9
Rows (pixels)	256	432
Columns (pixels)	256	432
Slice thickness (mm)	6	10
Slice spacing (mm)	6	11

Table 7-1 - Anatomical planning sequence parameters by manufacturer

7.3.3 Neural network design and training

The neural network design chosen was a modified version of the HRNet architecture,¹⁰⁵ which receives single-channel (grayscale) images and outputs 9 feature maps, corresponding to the possible identities of each pixel (aorta, left atrium, left ventricular wall, left ventricular cavity, pulmonary artery, pleural effusion, right atrium, right ventricular cavity) or a final ‘other’ (background) class. The network was trained until the validation loss plateaued (23 epochs), and the training process was augmented with random rescaling, rotation, shearing and translation. Loss was calculated over batches of 20 images by using the categorical cross-entropy loss function, and weights were updated using the Ranger optimizer (a combination of RAdam¹⁰⁶ & Lookahead¹⁰⁷). Programming was performed using the Python programming language with the Pytorch machine learning framework.⁶⁴ Training was performed on 2 GeForce RTX Titan graphical processing units (Nvidia, Santa Clara, California).

7.3.4 3D heart model reconstruction

The segmentation predictions of the neural network were converted into predictions of parameters such as cardiac chamber sizes by analysing each slice within the axial anatomical planning sequence and then re-assembling them into a 3D heart model (Figure 7-2). Specifically, each slice was fed into the neural network to yield a prediction of the identity of each pixel (left ventricle, pleural effusion, etc.). The slices were then resized according to the viewport information embedded in the DICOM file and the thickness of the slice, before combining the stacked series of slices into a 3D model through trilinear interpolation. This model was then analysed by calculating the volume of each structure present to give a value in ml. These were then scaled for body surface area to yield a value in ml/cm³. Finally, these volumes were adjusted to the “final” analysed volumes by fitting a linear regression model separately for each scanner manufacturer: this allowed a bias to be introduced to account for the different acquisition

parameters between the two manufacturers. Models were fitted for estimating LVEDVi, RVEDVi, LVMi, BSA-indexed LV mass-to-volume ratio, and BSA-indexed ascending aortic diameter.

For each measure, published BSA-indexed cut-offs were used to classify abnormal cases:

LVEDVi > 94ml/m² for LV dilatation, RVEDVi > 98 ml/m² for RV dilatation, LVMi > 83.5 g/ml for LV hypertrophy,¹⁰⁸ and > 0.84 for an abnormal LV mass-to-volume ratio.²⁷

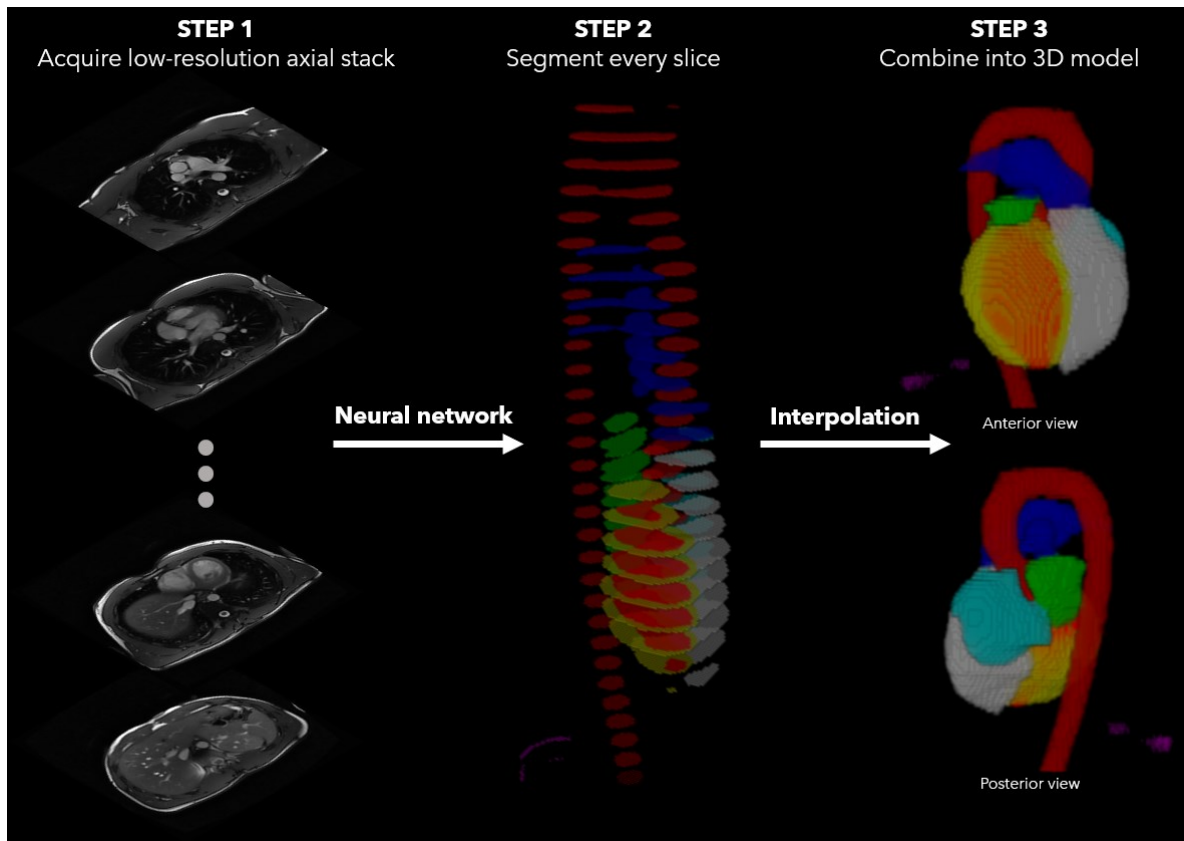


Figure 7-2 - Reconstruction of a 3D heart is possible by segmenting each slice within the axial anatomy stack. The slices can be combined through trilinear interpolation into a 3D model of the cardiac chambers and great arteries. This model can be used to make anatomical measurements, such as end-diastolic volumes and vessel diameters.

7.3.5 Statistical analysis

The segmentation performance of the neural network was assessed using the Dice coefficient, defined as:

$$Dice = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

Where for each class, TP refers to true positives (pixels classified by both the network and expert as of that class), FP as false positives (pixels classified by the network as that class, incorrectly) and FN as false negatives (pixels classified by the network as not that class, incorrectly).

The diagnostic performance of the network was reported for continuous outcomes using Pearson's r , along with the accuracy, sensitivity and specificity. Cohen's Kappa (κ) was also reported as a measure of accuracy resistant to class imbalance.

For the diagnosis of pleural effusions, there was no available threshold to use to convert a segmented volume (in ml) to a diagnosis of a pleural effusion. It is very common for patients to have a small trace of physiological pleural fluid visible posteriorly during a scan, and this would not be expected to be commented on in the radiology report. Indeed, what amount of pleural fluid warrants the diagnosis of a pleural effusion is likely to vary across radiologists and possibly the clinical question and other examination findings. Given this, I chose to report the performance as a binary classifier using a continuous predictor, specifically using both (1) logistic regression using and odds ratio, and (2) a receiving operating characteristic (ROC) curve with its associated area under the curve (AUC).

Statistical analysis was performed using R software (R Foundation, Vienna, Austria).⁶⁵

7.4 Results

7.4.1 Dataset

The baseline characteristics across the training and validation dataset and the testing dataset are shown in Table 7-2.

1500 slices underwent segmentation labelling, 1156 of which were within the training set and 344 within the validation set. Table 7-3 outlines the contents of the slices.

7.4.2 Segmentation performance

The mean Dice coefficient across all categories for the testing set was 0.910. The category-wise Dice coefficients are shown in Table 7-3 and ranged between 0.809 for the LV cavity to 0.929 for the aorta.

Examples of the human-supplied labels and the predictions of the neural network are shown in Figure 7-3.

	Training & validation set	Testing set
Number of studies / patients	375	200
Manufacturer		
Siemens	221	125
Philips	154	75
Male	229 (61.1)	115 (57.5)
Age (years)	55.7 (17.8)	56.7 (17.5)
LVEF (%)	55.6 (17.5)	55.7 (17.8)
LVEDV_i (ml/m²)	81.4 (34.6)	81.0 (38.1)
RVEDV_i (ml/m²)	80.1 (25.9)	78.8 (26.6)
LVM_i (g/m²)	77.6 (30.8)	76.4 (33.2)
Ascending aorta (mm/m²)	17.1 (4.2)	16.8 (4.1)
HCM	25 (6.7)	10 (5.0)
DCM	22 (5.9)	10 (5.0)
Pleural effusions	88 (23.5)	42 (21.0)

Table 7-2 - Characteristics of the included studies in the training and testing sets. Values are mean (standard deviation) for continuous variables, and count (percentage) for categorical variables. LVEF = left ventricular ejection fraction. LVEDV_i = Body-surface area-indexed left ventricular end-diastolic volumes. RVEDV_i = Body-surface area-indexed right ventricular end-diastolic volumes. LVM_i = Body-surface area-indexed right ventricular end-diastolic volumes. HCM = hypertrophic cardiomyopathy. DCM = dilated cardiomyopathy.

Feature	True presence within a slice		Network
	Training set	Validation Set	performance
	(n=1156)	(n=344)	(Dice coefficient)
Aorta	1059 (91.6)	316 (91.9)	0.929
Left atrium	404 (34.9)	116 (33.7)	0.925
LV cavity	558 (48.3)	151 (44.1)	0.809
LV wall	587 (50.8)	167 (48.6)	0.884
Pulmonary artery	182 (15.8)	65 (18.9)	0.907
Pleural effusion	347 (30.9)	128 (37.2)	0.890
Right atrium	481 (41.6)	139 (40.4)	0.924
RV cavity	616 (53.3)	178 (51.7)	0.910
Background	1156 (100.0)	344 (100.0)	0.995

Table 7-3 - Slices included in the training and testing sets. Values are n (%). The Dice coefficient reflects the segmentation accuracy of the neural network on the validation set.

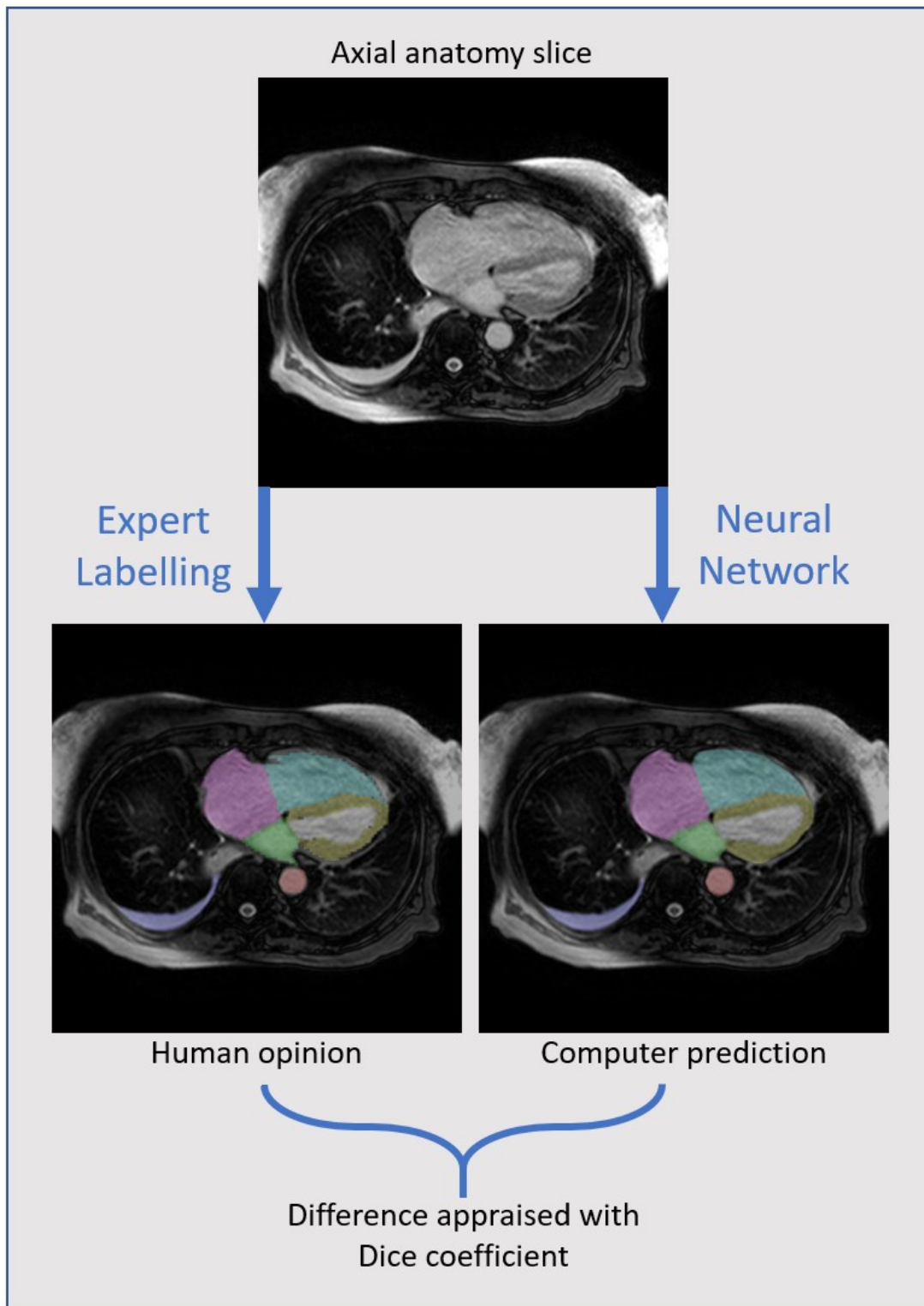


Figure 7-3 - A Human label and the AI's corresponding segmentation prediction. The figure shows a comparison between the human labels of a particular axial anatomical planning slice, and those predicted by the network for that slice.

7.4.3 Diagnostic model performance

By combining every slice from a scan's axial anatomic planning sequence, I was able to construct a 3D model of a patient's thorax (Figure 7-2) from which calculations could be made of chamber size but also for presence of abnormal features, such as pleural effusions.

BSA-indexed left ventricular end diastolic volume (LVEDVi) predicted by the network on the testing set correlated with the measures from the final report ($R^2 = 0.76$, $p < 0.0001$, Figure 7-4). The network was 90.5% accurate in identifying LV dilatation ($\kappa = 0.75$), with a corresponding sensitivity of 84.3% and specificity of 92.6%. Within the testing dataset, 10 studies were from patients with dilated cardiomyopathy, and the network correctly identified 9 (90%) of these as abnormal (95% CI 59.6% to 98.2%).

BSA-indexed right ventricular end diastolic volume (RVEDVi) predicted by the network on the testing set correlated with the measures from the final report ($R^2 = 0.53$, $p < 0.0001$, Figure 7-5). The network was 85.5% accurate in identifying RV dilatation ($\kappa = 0.62$), with a corresponding sensitivity of 80% and specificity of 87.1%.

BSA-indexed left ventricular mass (LVMi) predicted by the network on the testing set correlated with the measures from the final report ($R^2 = 0.74$, $p < 0.0001$, Figure 7-6). The network was 85% accurate in identifying LV hypertrophy ($\kappa = 0.64$), with a corresponding sensitivity of 78.9% and specificity of 87.4%. LV mass:volume ratio predicted by the network on the testing set correlated with the measures from the final report ($R^2 = 0.5$, $p < 0.0001$, Figure 7-7). The network was 71% accurate in identifying a raised LV mass:volume ratio ($\kappa = 0.41$), with a corresponding sensitivity of 68.9% and specificity of 74.4%. Within the testing dataset, 10 studies were from patients with hypertrophic cardiomyopathy, and the network correctly identified all (100%) of these as abnormal (95% CI 72.3% to 100%).

Only 54 of the 200 testing dataset cases provided ascending aorta diameters. BSA-indexed ascending aorta diameters predicted by the network on the testing set correlated with the measures from the final report ($R^2 = 0.82$, $p < 0.0001$, Figure 7-8). The network was 94.4% accurate in identifying ascending aortic dilatation ($\kappa = 0.79$), with a corresponding sensitivity of 100% and specificity of 93.6%.

Figure 7-9 demonstrates the relationship between the pleural fluid volume predicted by the neural network and the diagnosis of a pleural effusion made by the radiologist on the formal report. The blue line demonstrates the probability of a pleural effusion being present, according to a logistic regression model fitted on these data. The corresponding odds ratio was 1.06 (95% confidence interval 1.04 to 1.09; $p < 0.0001$). The corresponding ROC curve AUC was 0.906 (Figure 7-10).

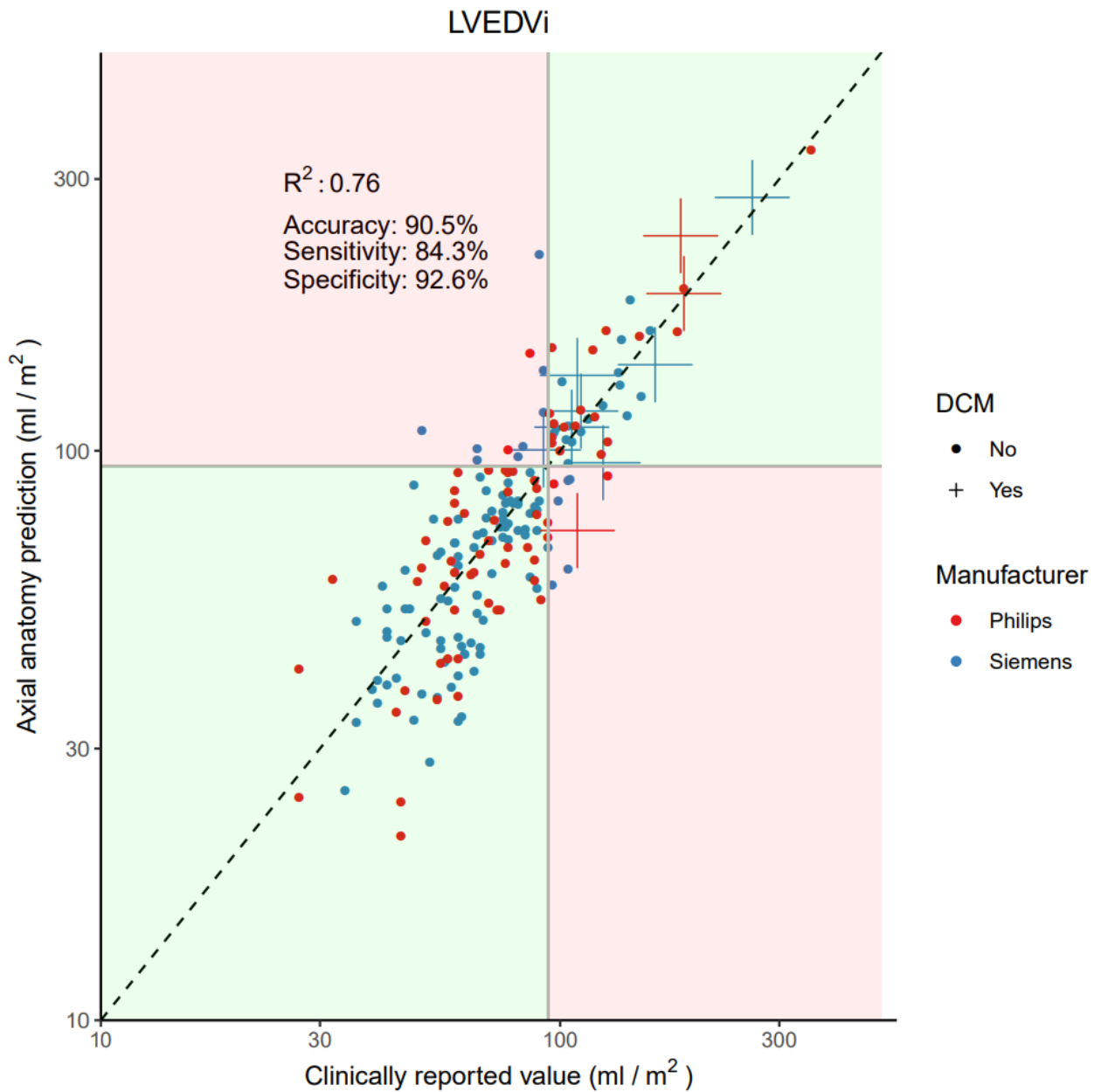


Figure 7-4 - The relationship between the neural network's chamber size predictions for body surface area-indexed left ventricular end-diastolic volume (LVEDVi) versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively.

Patients with known dilated cardiomyopathy (DCM) are shown as crosses.

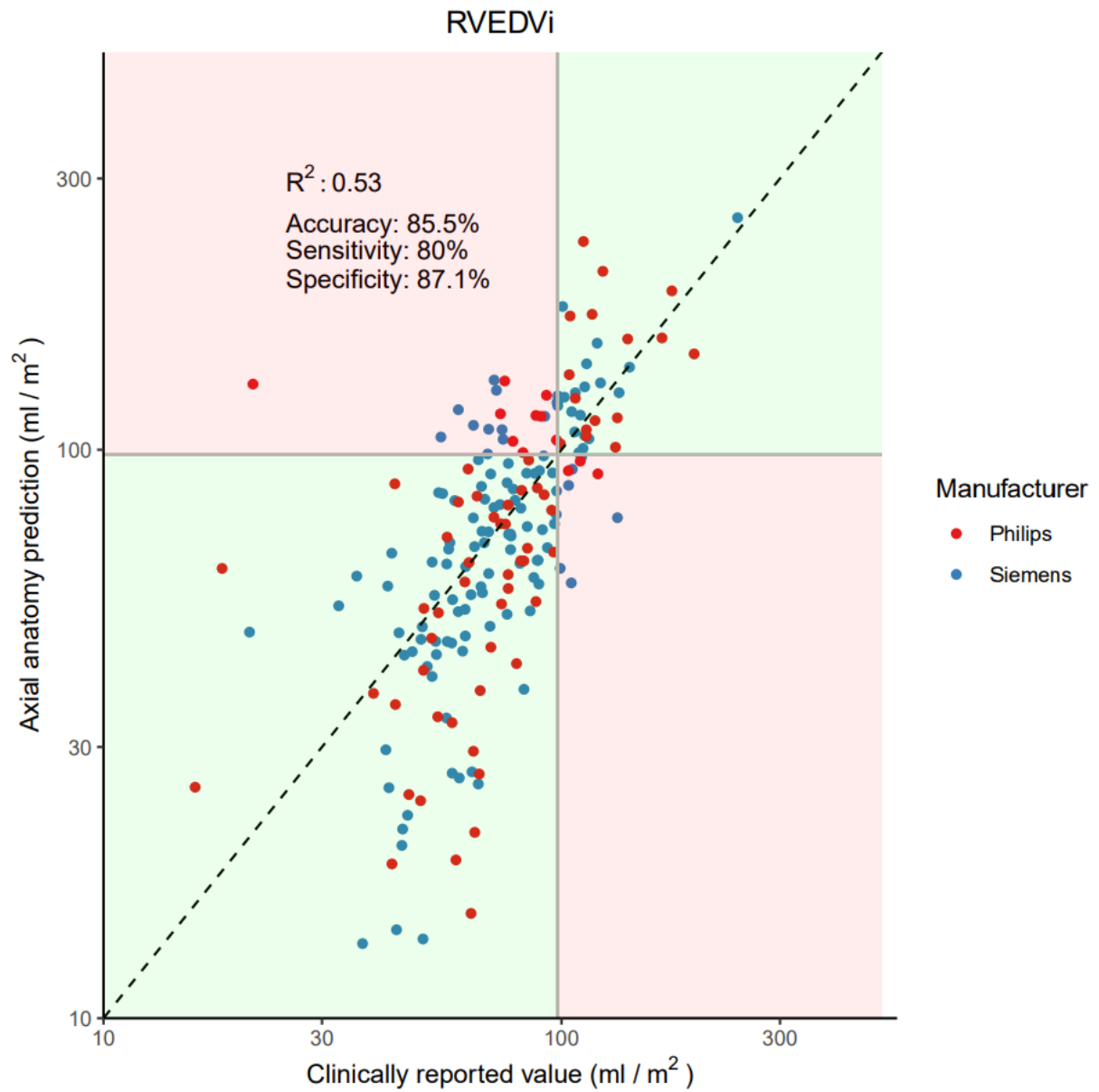


Figure 7-5 - The relationship between the neural network's chamber size predictions for body surface area-indexed right ventricular end-diastolic volume (RVEDVi) versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively.

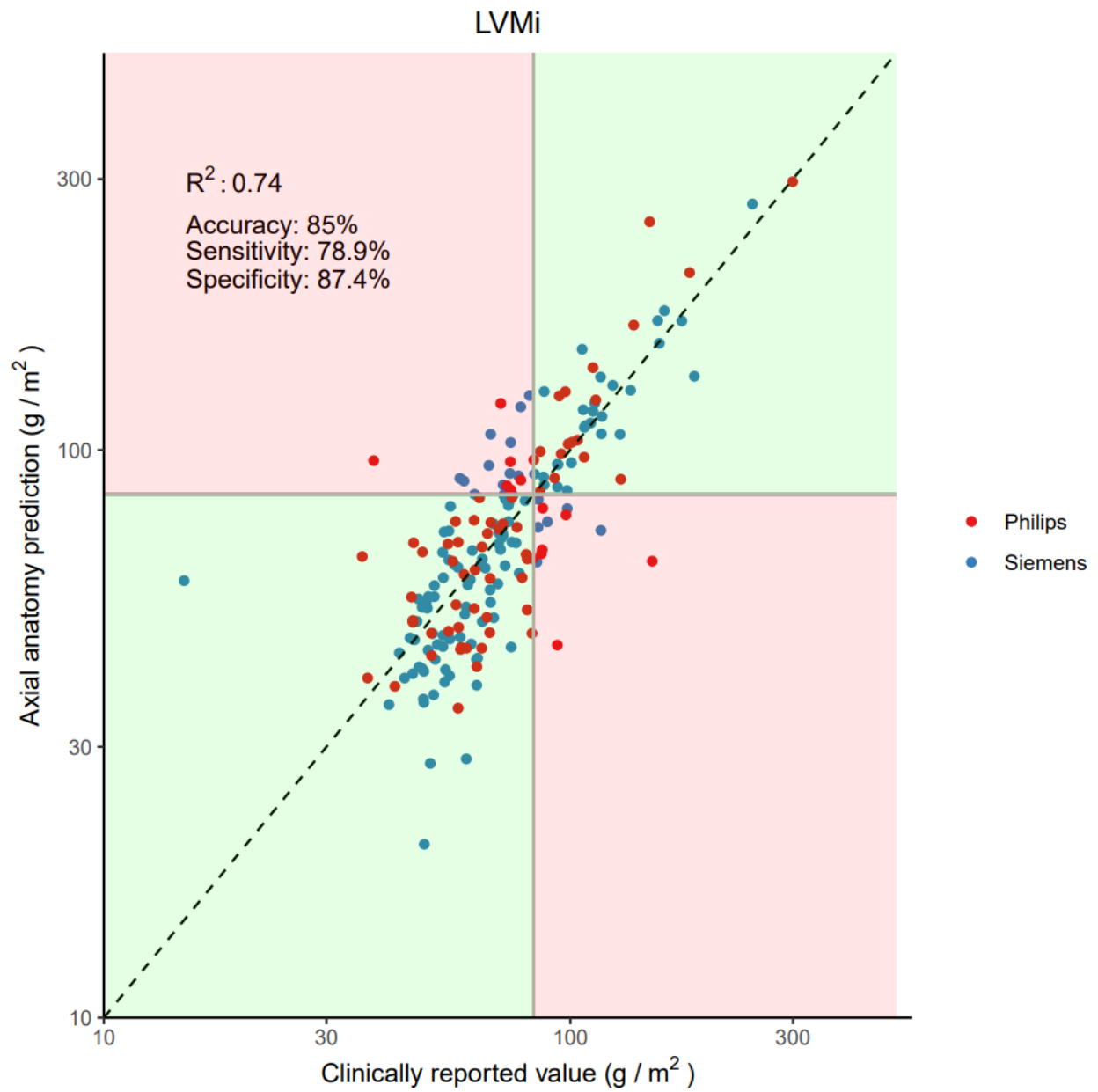


Figure 7-6 - The relationship between the neural network's predictions for body surface area-indexed left ventricular mass (LVMi) versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively.

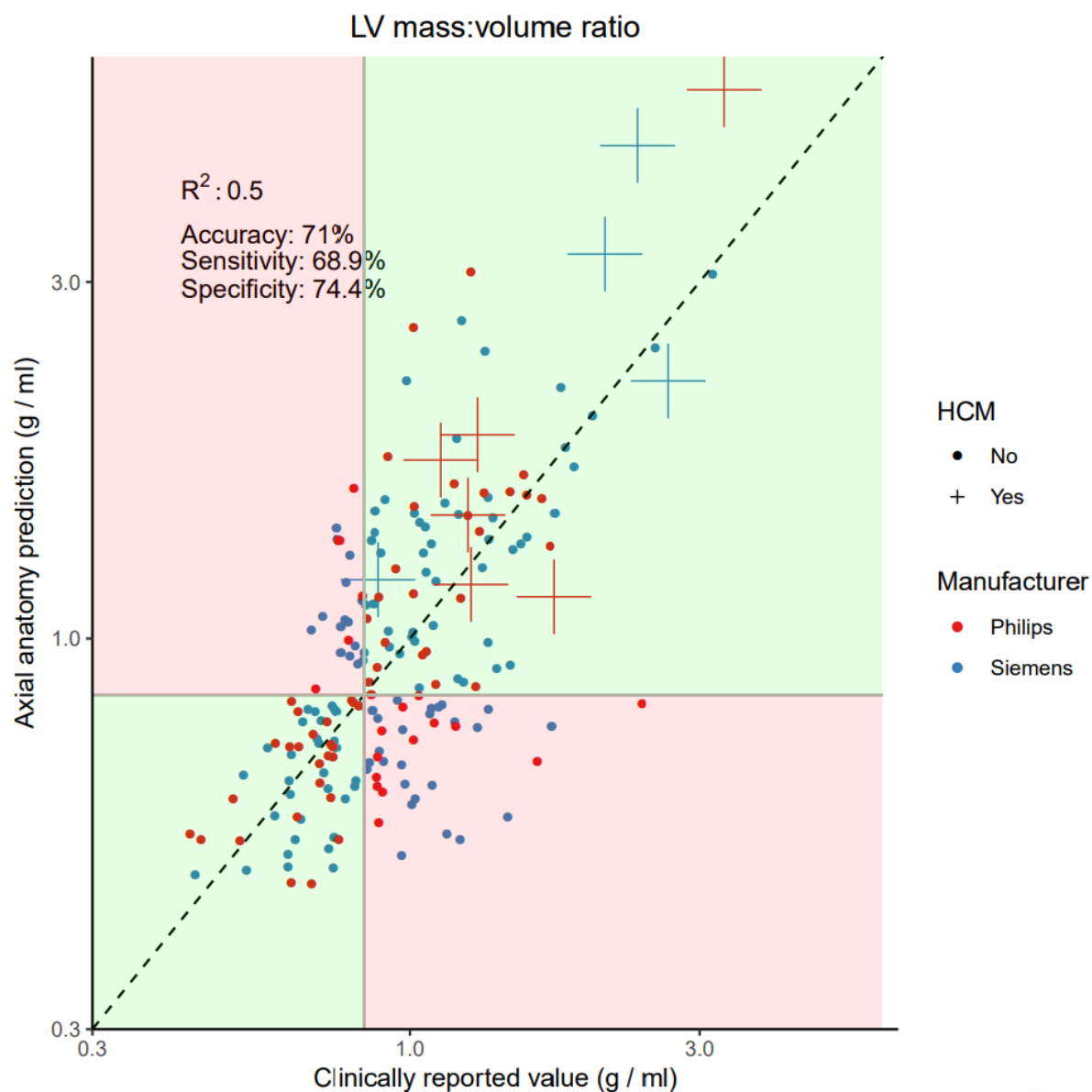


Figure 7-7 - The relationship between the neural network's predictions for left ventricular (LV) mass to volume ratio versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively.

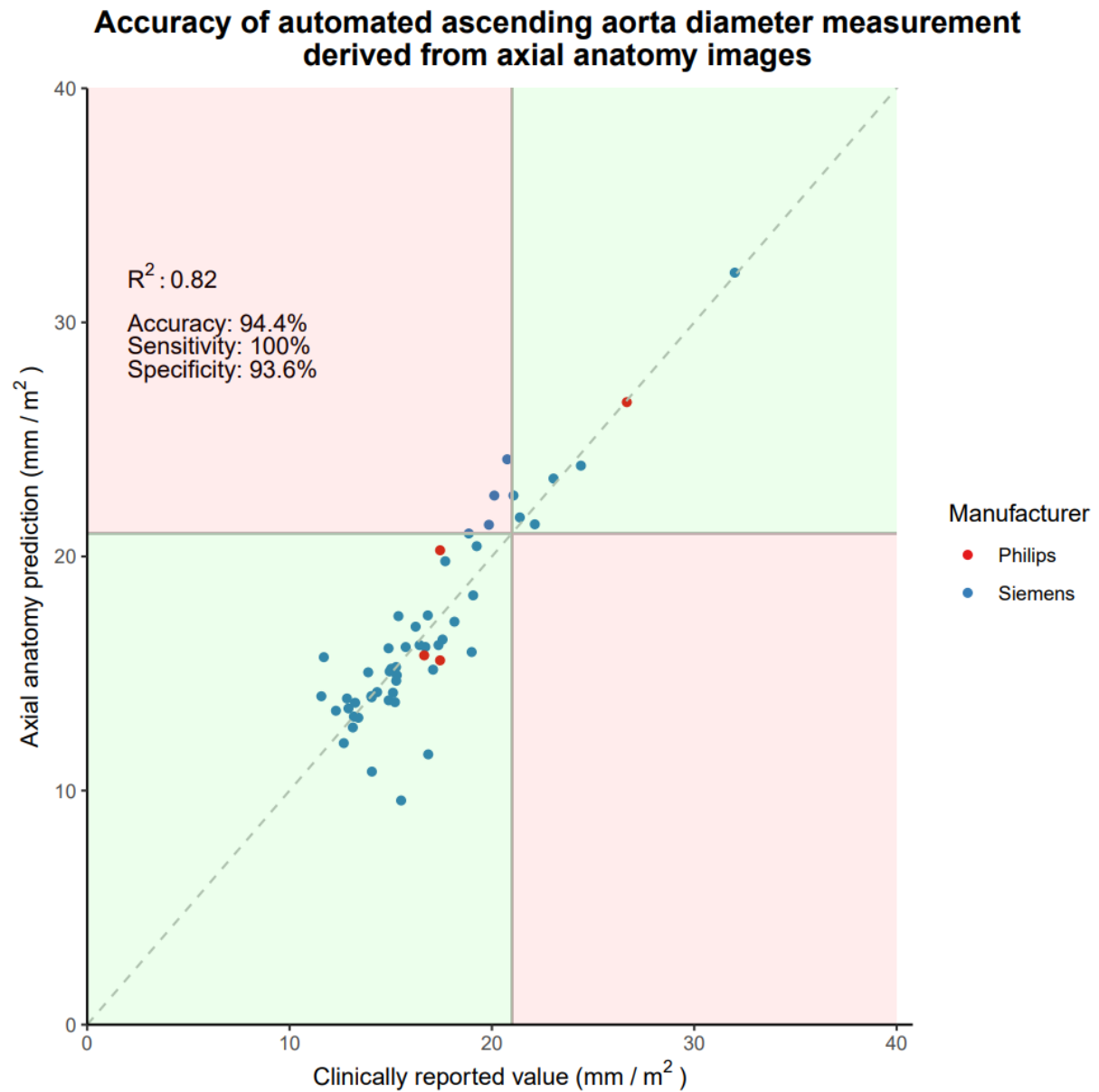


Figure 7-8 - The relationship between the neural network's body surface area-indexed ascending aortic diameter predictions versus those extracted from the final report. Green zones and red zones symbolise areas of the plot where the neural network agrees and disagrees with the values from the full scan, respectively.

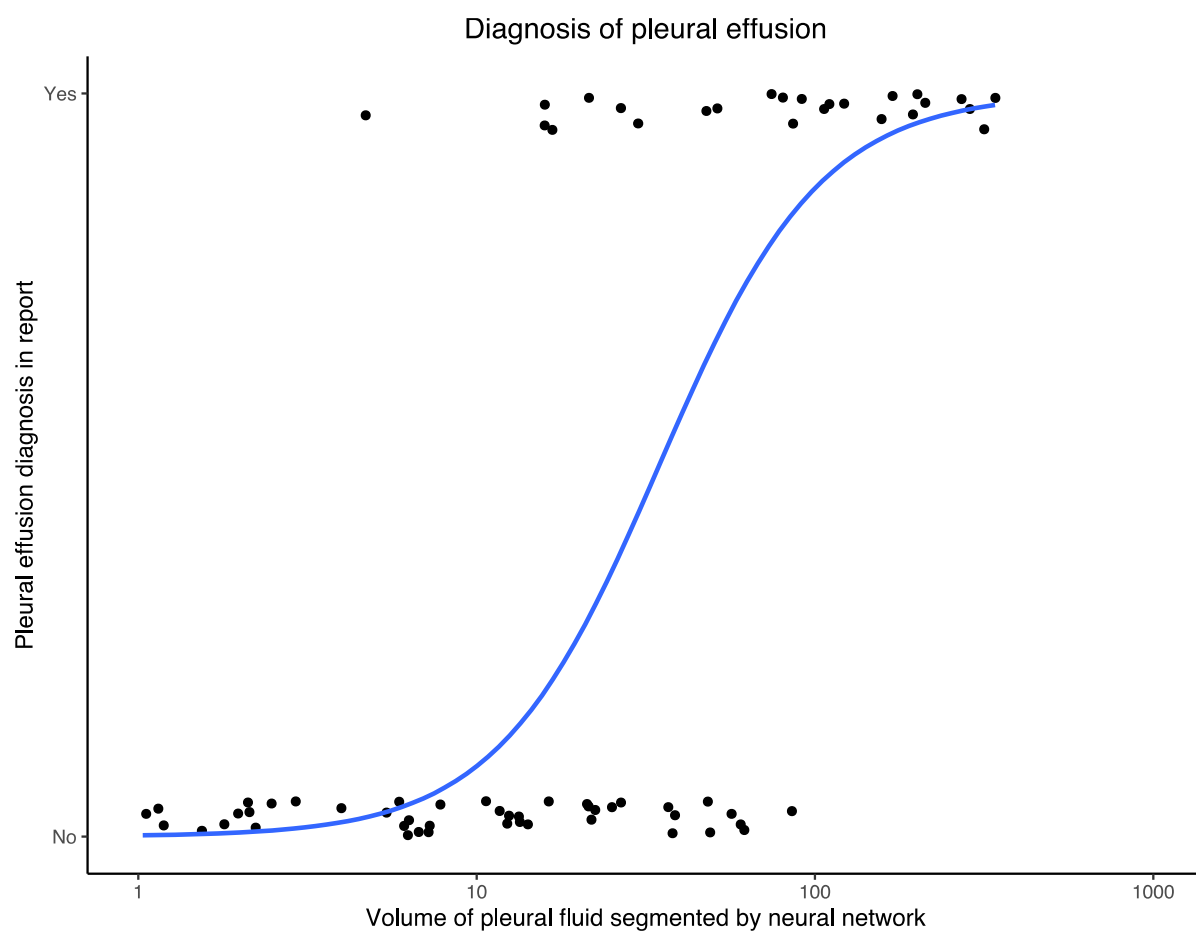


Figure 7-9 – The relationship between the predicted pleural fluid volume and the diagnosis of a pleural effusion. The blue curve represents the logistic regression model fitted.

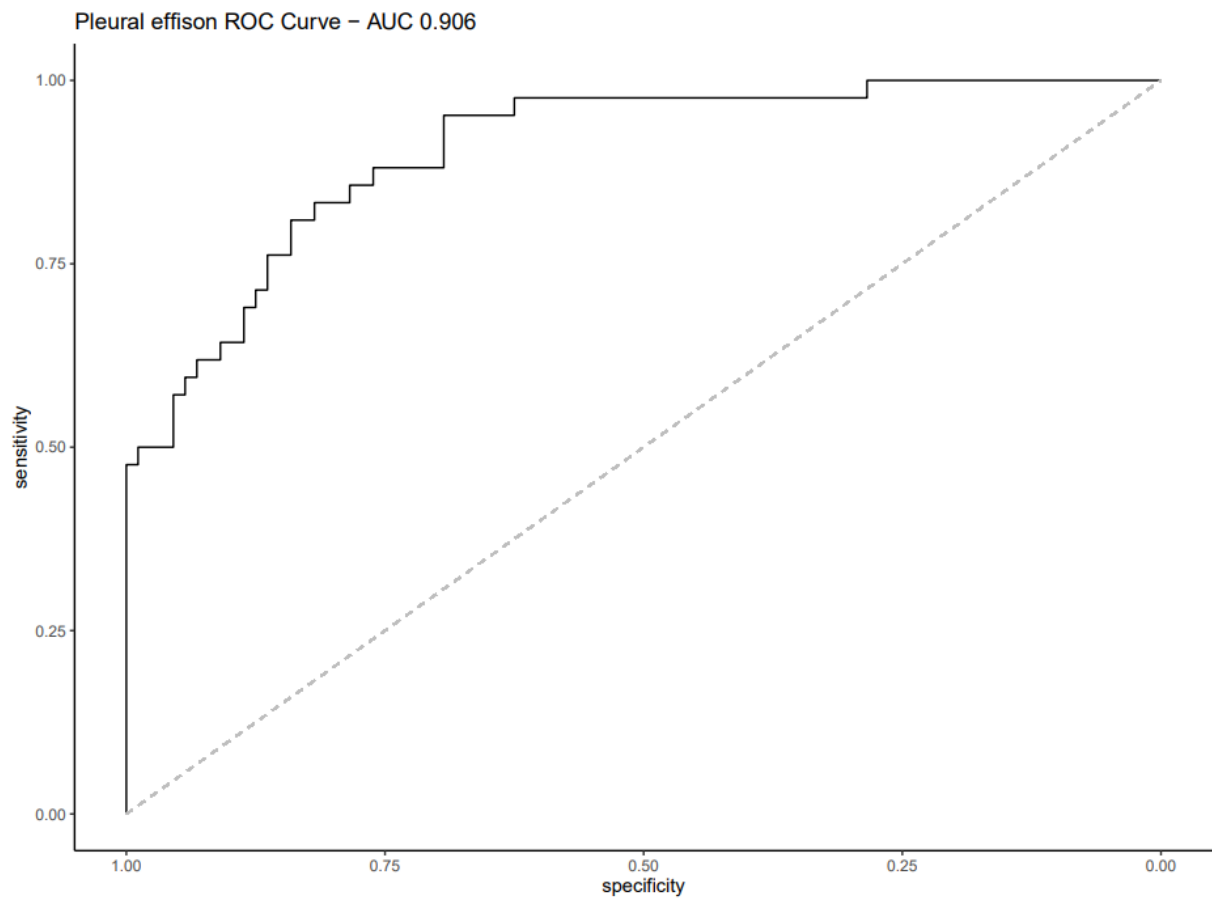


Figure 7-10 - Receiver operating characteristic (ROC) curve for pleural effusion identification. The area under the curve (AUC) is 0.906.

7.5 Discussion

This study demonstrates that accurate important diagnostic information can be derived from axial anatomy images obtained at the start of a CMR scan. These results are consistent across two different scanner manufacturers. This system could allow technicians performing the scan to be signposted to unexpected pathology and help direct optimized image acquisition for the remainder of the scan.

7.5.1 Earlier & automated diagnosis – a comparison with previous approaches

Neural networks are now rivalling and surpassing humans for cardiac chamber segmentation and quantification^{102,103}, in the setting where images of the cardiac chambers are acquired in a dedicated conventional manner in patient-specific scanning planes, adjusted by the radiographers. However, the aim of this work was different in two ways.

First, the network in this study has the potential to identify extra-cardiac diagnoses such as ascending aorta dilatation and pleural effusions. Being able to identify these findings may allow an adaptive approach to scanning protocols which avoids recalling patients for additional images and even gives technicians performing the scan reassurance that additional images are not required.

Second, previous studies have aimed to improve absolute quantification of chamber size and function and have therefore been trained to process the high-quality cine imaging which reporting clinicians currently use. Such sequences differ from the anatomical stack generated and processed in this paper in several important ways.

First, the dedicated sequences' scan planes are ideally orientated with respect to the patient's heart to ensure radial function is within rather than through the plane of imaging. This minimises the risk of the phenomenon termed 'partial voluming', where a slice contains two different types of tissue (e.g. blood and myocardium) at different levels within its depth. Second, the dedicated sequences are acquired with less spacing, allowing more samples per ventricle. Third, the dedicated sequences take considerably longer to acquire: they are acquired over 8-15s per slice, of which there may be up to 10 slices per sequence.¹⁰²

In contrast, the entire axial anatomical sequences can be acquired in 3-4 breath holds. However, they are of relatively large slice thickness, and are orientated axially with no reference to the patient's heart. Given this, the ability to acquire accurate diagnostic information early on in the scan has the potential to triage the application of further sequences during the scan. For example, patients found to have LVH and pleural effusions on axial anatomy images may have a diagnosis of cardiac amyloid in which pre-contrast T1 maps would be useful.¹⁰⁹ As another example, dedicated aortic imaging is too laborious for routine use on each patient but could be reliably targeted to those who need it using this method.

7.5.2 Implications for reporting prioritization, supervision and patient safety

A system providing automated diagnosis during the earliest stages of a cardiac MRI scan would not only be useful for ensuring scans are correctly protocolled but would also allow physicians to prioritize the supervision and reporting of abnormal scans.

Patients with pleural effusions, for example, are particularly at risk of respiratory difficulties when lying flat for extended periods of time, as necessitated by a CMR scan. Their identification at the earliest stages of could increase vigilance of these more vulnerable patients, and facilitate

the alteration of sequences to allow shorter breath holds, motion-corrected free-breathing sequences and accelerated protocols to minimize scan time.

Scans shown to contain unexpected pathologies may also be flagged for expedited review and reporting by physicians. For example, an outpatient screening CMR scan in a low risk patient might be identified as demonstrating unexpected marked LV dilatation with pleural effusions. Such patients at risk of decompensation and could be identified for early follow-up.

7.5.3 Study limitations

The neural network described in this study is not 100% accurate. Even the most accurate measurement (ascending aortic diameter) is only 94% accurate on our dataset. However, the correlation between the neural network's predictions and the true measurements for the measures examined ensure that the extreme biological values associated with disease are more consistently correctly identified by the network (100% of hypertrophic cardiomyopathy cases, 90% of dilated cardiomyopathy cases).

Furthermore, previous studies have shown the coefficient of variation is over 10% for estimating left ventricular volumes, even when assessed by the same doctor in the same patient using dedicated left ventricular sequences.¹⁰² In this study, we have compared the performance of this network against human observers behaving clinically, and therefore this 10% variation inherently sets an upper ceiling on the correlation coefficient obtainable by the network – it cannot correlate with the human observers better than the human observers correlate with themselves.

With all deep learning studies, there is concern that the findings in this study may not generalize to a wider population.¹¹⁰ This can be due to a phenomenon of “overfitting”, where the neural network is highly accurate at processing images on which it was trained but performs much less well on unseen “real world” examples. To try and mitigate these concerns, I took several

approaches. First, the performance is reported on a test set which was only assessed after training the neural network. Second, the dataset we assembled was from two different hospitals across multiple reporting physicians. Third, the dataset comprises scans across both Siemens and Philips scanners, with the correlation plots showing similar accuracies for both manufacturers. Finally, I have released the neural network online, so that its performance can be assessed by any interested academic or clinician.

7.6 Conclusion

This Chapter has demonstrated that a neural network can accurately reconstruct a 3-dimensional model of the heart and major vessels from the very first sequences of a CMR study. This system is able to accurately quantify cardiac chamber size, aortic diameter and presence of pleural effusions. I have made trained neural networks publicly available for use.

8 Synthesis

In this thesis I have investigated the role of convolutional neural networks (CNNs) in four different modalities of cardiac investigations: X-rays in chapter 4, electrocardiograms (ECGs) in chapter 5, echocardiography in chapter 6, and cardiac magnetic resonance imaging (CMR) in chapter 7.

Whilst these networks all share the property of being convolutional neural networks, they need different designs to tackle the different forms of data. This results in different detailed architectures. However, what has made convolutional neural networks so successful for analysing images and data streams is that they are able to recognise patterns that may have different temporal or spatial locations without having to have a specific detector for each of the myriad possibilities. It is that breakthrough that makes the convolutional design unrivalled.

I have utilised CNNs in both classification tasks (chapters 4, 5 and 6) and segmentation tasks (chapter 7).

I have implemented networks which are 1-dimensional (chapter 5), 2-dimensional (chapters 4, 6 and 7) and 3-dimensional (chapter 6).

I have developed CNNs that outperform current algorithmic methods for established problems (chapter 4). I have developed CNNs that improve on previously published CNN approaches (chapter 6). I have developed CNNs to deal with novel clinical problems (chapter 5). And finally, I have developed CNNs which are able to process previously acquired data in novel ways (chapter 7), aiming to improve healthcare delivery and patient experience.

In this integrative chapter I will address several important concepts arising from the thesis as a whole. Finally, I will conclude with a discussion of how AI may influence our practice in the future, and the implications for the medical workforce.

8.1 Convolutional neural networks provide options for automation of tasks which are difficult to solve with traditional algorithmic approaches.

Traditional algorithmic approaches, where a sequence of steps is explicitly programmed by a human, have become commonplace in modern medical practice. Numeric risk scores such as Q-RISK2¹¹¹ have become part of routine care, and decision trees exist for classifying recognised ECG features, such as for locating accessory pathways.¹¹²

Both ECGs and X-rays were discovered in 1895. For decades, there have been computer algorithms to generate usable ECG reports.⁸⁴ However, no such traditional algorithmic system has even been successfully applied to X-rays. The reason for this is the vast complexity of the task of computer vision if tackled using algorithms, i.e. steps carefully specified by human programmers. The ECG is just within the limits of the complexity of data can be analysed in this way, and it is still subject to embarrassing machine errors. For the X-ray, performance using an algorithm would be so laughable as to prevent any vendor from daring to offer it.

Convolutional neural networks have revolutionised computer vision. Over the last decade, ImageNet, a database of over 14 million images, has been used to assess the relative performance of different image classifiers.¹¹³ AlexNet, the first successful deep convolutional neural network, brought the top 5 error rate down from 26% to 16% in 2012 (Figure 8-1).⁵⁵ Just 5 years later, CNNs had reduced this to just 2.25%,¹¹⁴ making half as many mistakes as humans.

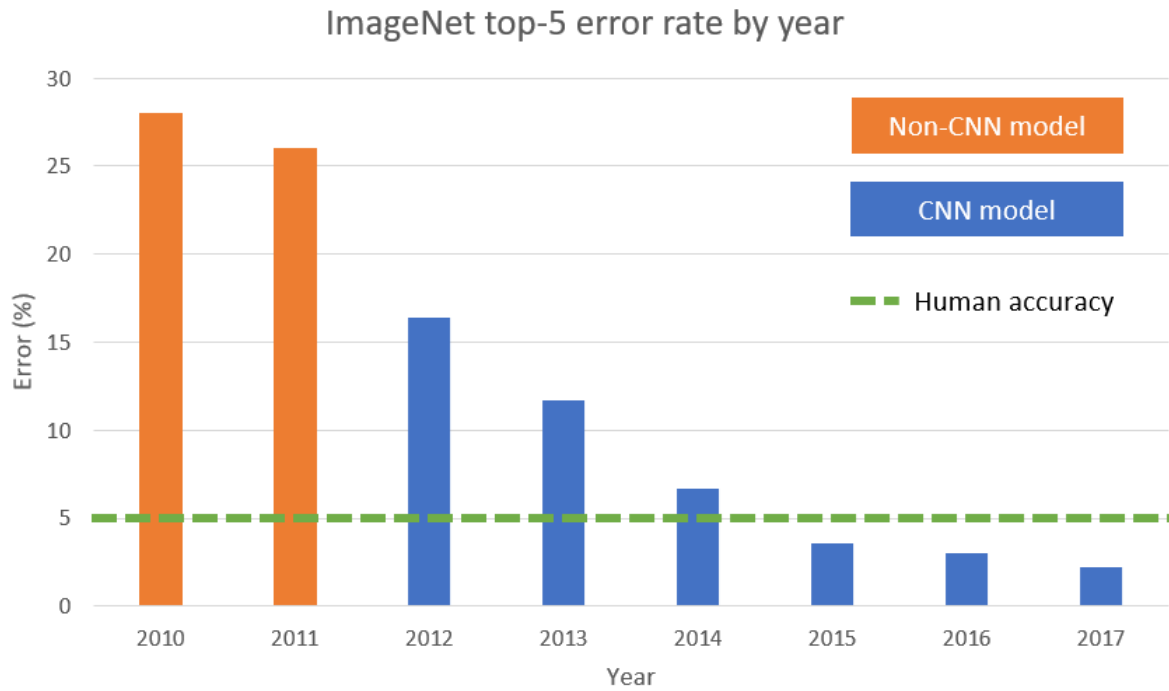


Figure 8-1 - Accuracy on the ImageNet image classification challenge by year. Data from <http://www.image-net.org/>. In 2012, the AlexNet convolutional neural network (CNN) posted a dramatic reduction in the error rate above all previous approaches, and CNNs have consistently improved upon this performance since. In 2015, CNNs surpassed human performance (green dotted line).

CNNs now dominate these image classification (and segmentation) tasks because such tasks are traditionally exceptionally difficult to solve using traditional algorithmic methods. For example, imagine we are programming an algorithm with the aim of discriminating between cats and other animals. There are many features we might think useful, such as the presence of whiskers, the size of the animal, triangular-shaped ears, triangular shaped nose, furry coat, tail, four legs, and so on. Now imagine the small animal depicted in Figure 8-2. This is clearly a puppy, and yet it meets all of the previous criteria. Designing our algorithm so that it is confident that this image is *not* a cat may appear difficult. However, when we consider the animal may be facing *away* from the camera in some pictures, it immediately becomes apparent that manual programming of such complex classifiers is *incredibly* difficult. In this thesis I have developed CNNs for four different clinical problems, because CNNs allow us to develop classifiers by training *from example*.



Figure 8-2 - Is this a cat? Reproduced from <https://www.flickr.com/photos/27587002@N07/5170590074> under a CC BY 2.0 license.

In Chapters 4 (page 104) and 5 (page 133), I have described how I developed CNNs which are able to (1) identify which model of cardiac rhythm device is present on a chest X-ray, and (2) discriminate between the different ECG responses to His bundle pacing, respectively. These are a rare class of problem, because there are already in clinical use explicit algorithmic approaches.

In the case of cardiac rhythm device identification, a flow chart has been developed.⁵ This has been possible because cardiac rhythm devices have a very stereotypical appearance on X-ray; all devices of a certain model are completely identical, and the appearance of a device is predictable, even when it is back-to-front, due to the penetrating nature of X-rays. However, I demonstrate in chapter 4 that a CNN approach greatly exceeds the performance of the flow-chart, even when compared with the figures in the manual algorithm's index publication.

Similarly, complex algorithms exist to discriminate between the different ECG responses to His bundle pacing.¹² These involve assessing electrocardiograms recorded from the His pacing lead and several ECG features (such as the isoelectric segment and the paced QRS duration) and comparing these with either the patient's baseline ECG, or a paced ECG at a different voltage (which requires a pacemaker programmer to be used). In chapter 5 I developed a neural network with the aim of simplifying this process. Whilst the 75% accuracy of this proof of concept falls short of what would be required for a system that could replace expert humans in the pacemaker implantation laboratory, it is able to provide instantaneous beat-to-beat classifications using only a single ECG complex.

In this thesis, I have also investigated two applications of CNNs where there is currently no available algorithmic approach.

Automated classifications of echocardiograms, according to the view they depict, is a complex task for several reasons. First, there is no colour information which can be used to identify structures. Second, the brightness of areas of an image may impart very little information, with neighbouring structures being of the same average brightness, but differing only subtly in

texture. Third, neighbouring structures typically lack a distinct border; instead, each structure will usually have a slightly different texture, rather than there being a definitive boundary present. For these reasons, there were no adequate solutions for this task before the introduction of CNNs. Since then, previous groups have demonstrated the ability of CNNs to classify echocardiograms by their depicted view,^{25,26} but in Chapter 6 I demonstrate that previous solutions' reliance on individual frames limits their performance. The introduction of temporal neural networks allows movement of cardiac structures through times to be assessed by the network, as a human would (Figure 6-8, page 198).

Finally, I have demonstrated in Chapter 7 that a CNN is able to process a clinically disregarded three-dimensional dataset acquired at the start of each CMR scan: the axial anatomy stack. No previous algorithmic or CNN approaches exist for analysing these three dimensional off-axis images of the entire thorax. I demonstrate as a proof-of-concept that a CNN is able to accurately segment these images to provide estimates of ventricular volumes, left ventricular mass and aortic dimensions within minutes of a scan commencing.

8.2 Different problems are suited to different architectures of convolutional neural network

The scientific venture of machine learning is not merely the dumping of data into a standard piece of software. Clinical researchers within the field of machine learning are sometimes considered to be carrying out a trivial task. One of my BSc students received the following comment and a low mark for his project:

“Student given very easy access to data acquisition – essentially just needed to load ECGs from a database, label P, QRS, T wave and press play on the Neural Network analysis package”

Although this was disappointing, I reassured the student that this was due to failure to understand that the work of the machine learning scientist ends, rather than begins, at the point that we ‘press go’. This may not be obvious to researchers in fields where the labour is primarily during the experiments, rather than before them.

I started with the assumption that the more complex and advanced the network was, the better it would perform. In two-dimensional image classification there have been a wide range of neural network architectures created, and there is excellent evidence demonstrating the relative performance of these architectures on the ImageNet dataset.¹¹³ It might therefore appear sensible to utilise the best performing of these when applying convolutional neural networks to new tasks.

However, I was concerned that the data used in my thesis are so different from images in the ImageNet database (which is made up of colour images of everyday objects), that an architecture’s performance on ImageNet may not be a reliable indicator. For example, X-rays and echocardiograms comprise greyscale rather than full colour data, and ECG analysis should be performed with 1-dimensional rather than 2-dimensional convolutional layers.

For this reason, across the first three results chapters I have investigated the relative performances of different neural networks.

In chapter 4 I investigated the efficacy of 5 different neural network architectures at identifying the model of cardiac rhythm device present on an X-ray. Table 4-2 (page 118) demonstrates the accuracy of these networks, and interestingly shows that the most modern and best-performing network in terms of ImageNet, DenseNet121, performed less well than a design over a year older: Xception. Xception also outperformed models with more parameters, such as ResNet50. Furthermore, one network, VGGNet, failed to learn to perform this task at all.

Similarly, in chapter 6 I investigated the ability of neural networks to classify echocardiograms, and I assessed a much wider variety of neural network architectures. However, I did find a similar pattern of performance, with Xception again outperforming the other newer and larger 2-dimensional networks.

In chapter 5, I took on a different task from classifying 2-dimensional images: classifying waveform data in the form of ECGs. In this task I implemented 16 different neural network architectures, including both current state-of-the-art neural networks for ECG classification, and a variety of other networks which I adapted from the image classification field. Interestingly, I found a relatively small neural network, comprising only 1.8 million parameters, outperformed neural networks more than 10 times its size, as well as the previously published ECG-specific architectures.

The findings across these chapters illustrate an important point – research progress outside of the medical field does not appear to necessarily translate to medical images, and the best neural network for a task appears to be problem-specific. I therefore argue that studies which rely on a single “state-of-the-art” neural network architecture from another field would benefit from appraising a wider variety of architectures, as differences in accuracies can be marked (accuracy of 4.4% versus 91.1% for the worst- and best-performing networks in chapter 4; accuracy 37.7% versus 75% for the worst- and best-performing networks in chapter 5).

8.3 We now have many good neural networks: the bottleneck for future medical AI will be capturing appropriate expert labelling

I have attempted four different machine learning tasks during my PhD, and all four have been successful, to varying extents.

However, during each of these projects, the most time-consuming task has been the labelling of the large amounts of data required. In Chapter 4 I had to manually crop 1,676 X-rays of pacemakers. In Chapter 5, my colleague had to manually segment and classify 1,437 ECGs. In Chapter 6, I classified 9,098 echocardiogram videos. Finally, in Chapter 7, I segmented each of the cardiac chambers and great vessels in 1500 cardiac MRI slices from 375 studies. Some of these tasks were not only time consuming, but the specialist knowledge required to perform them precluded using lay volunteers to perform the task, through online services such as Amazon's Mechanical Turk (<https://www.mturk.com/>, Amazon Inc., Seattle, WA, USA).

For this reason, during my PhD, I devoted significant time developing bespoke labelling software to improve the speed at which I could generate high quality labelled data.

In Chapter 5 I developed software which allowed my colleague to easily segment ECGs from electrophysiology cases (Figure 5-2 and Figure 5-3). Systems such as this are commonplace in the field of image processing, but no such system existed which would allow rapid labelling of 12 lead ECGs. This software works across both Windows and Mac computers and is fully open-source, and available at <https://github.com/jphdotam/BARD-labeler>. Because of its modular and configurable design, I have received messages from international collaborators who now use this software to segment ECGs. We are currently using this software on ongoing AI projects, such as predicting the locations of abnormal electrical locations within the heart using ECGs.

In Chapter 6 I developed a truly novel labelling system, which I used to classify echocardiograms according to the view depicted. This system is based around two principles:

8.3.1 Multiple stages of fewer choices are more efficient

First, for humans, the task of classifying each echocardiogram is made easier when the number of options is reduced. Instead of asking humans to choose 1 of 14 separate views for each video and presenting them with 14 option buttons to navigate through, the human first classifies each video according to a more simple distinction: where the probe is located on the patient's chest. Then, once all videos have been classified as one of these five views (Figure 6-4), the videos are then further classified according to the exact view shown (Figure 6-5). This greatly simplifies the task, as it reduced the total number of options (and mistakes) I was presented with, and allows keyboard number keys to be used instead of the mouse, as double-digit IDs are avoided. This system allowed a second expert to label the testing dataset of 2140 videos in 4 hours (an average speed of 6.7 seconds per video).

8.3.2 Spotting odd ones out is even more efficient

The second principle of the software was computer assisted labelling. Once I had labelled several hundred videos, I found I could train a neural network with sufficient accuracy that it could propose putative labels for me, that I could then rapidly view. Figure 6-6 shows this system in use, where a grid of 12 videos that the computer believes depict apical 4 chamber views (A4CH) are shown playing. The third image on the top row was found to be incorrectly classified by the network, and the human then flags this for manual relabelling at a later date. This system allowed me to label over 9000 echocardiogram images, over 7000 of which were proposed by the network, and approved quickly by myself.

These examples have demonstrated to me the importance of developing highly efficient and user-focussed labelling software, and this is now translating into my post-doctoral research. I have since developed bespoke cardiac MRI segmentation software (Figure 8-3) which is currently

being used by researchers both at Imperial College London, and the National Institute for Health in Baltimore, USA. I am also working with other members in our group to develop a truly portable, web-based labelling system which will allow international collaborators to label echocardiograms, without needing to install any specialist software. This software is shown in Figure 8-4 and is able to function on both traditional computers as well as smartphones and tablets.

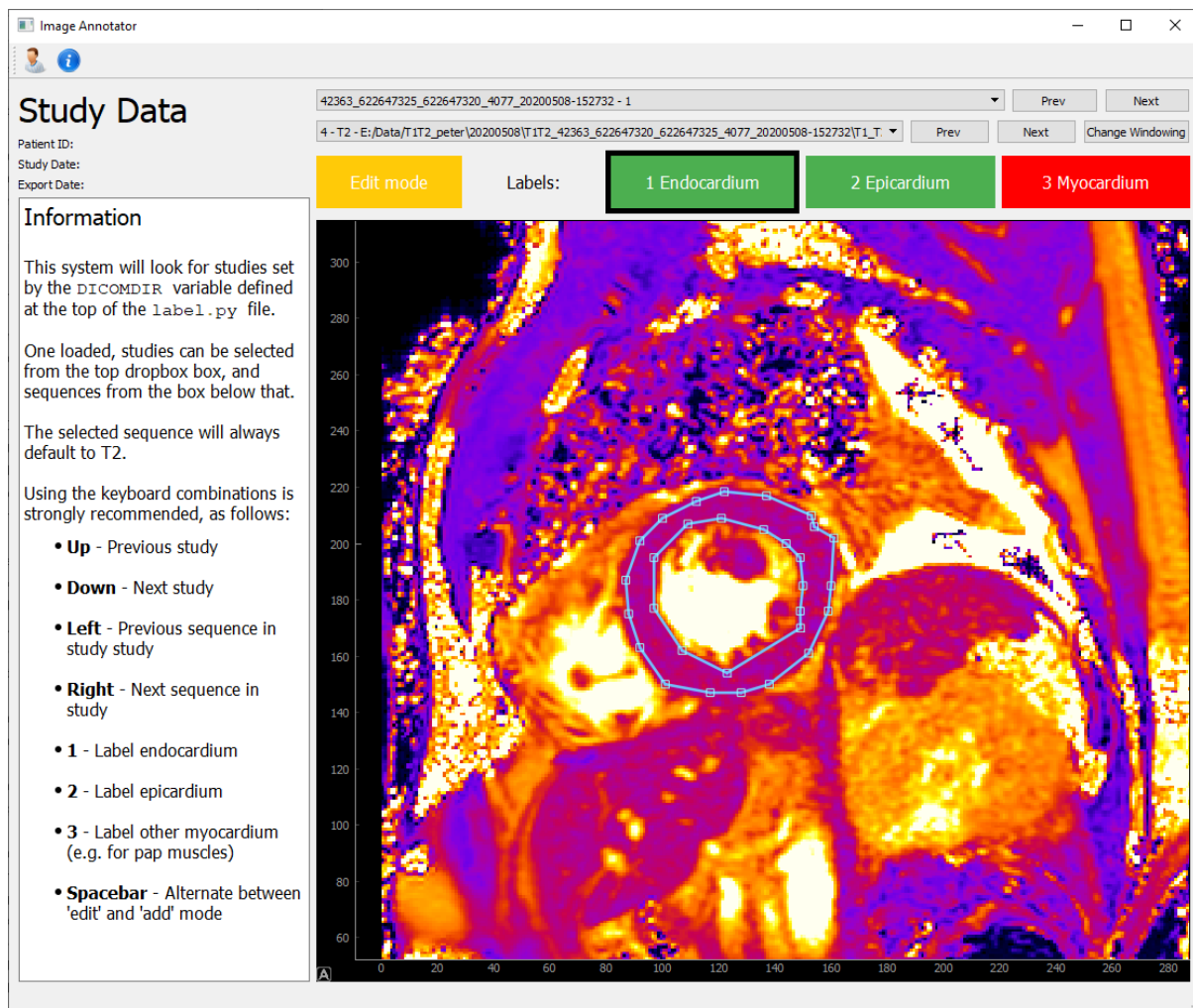


Figure 8-3 – Labelling software I have recently developed for an ongoing research project, involving the segmentation of the endocardial and epicardial left ventricular surfaces on cardiac MRI images.

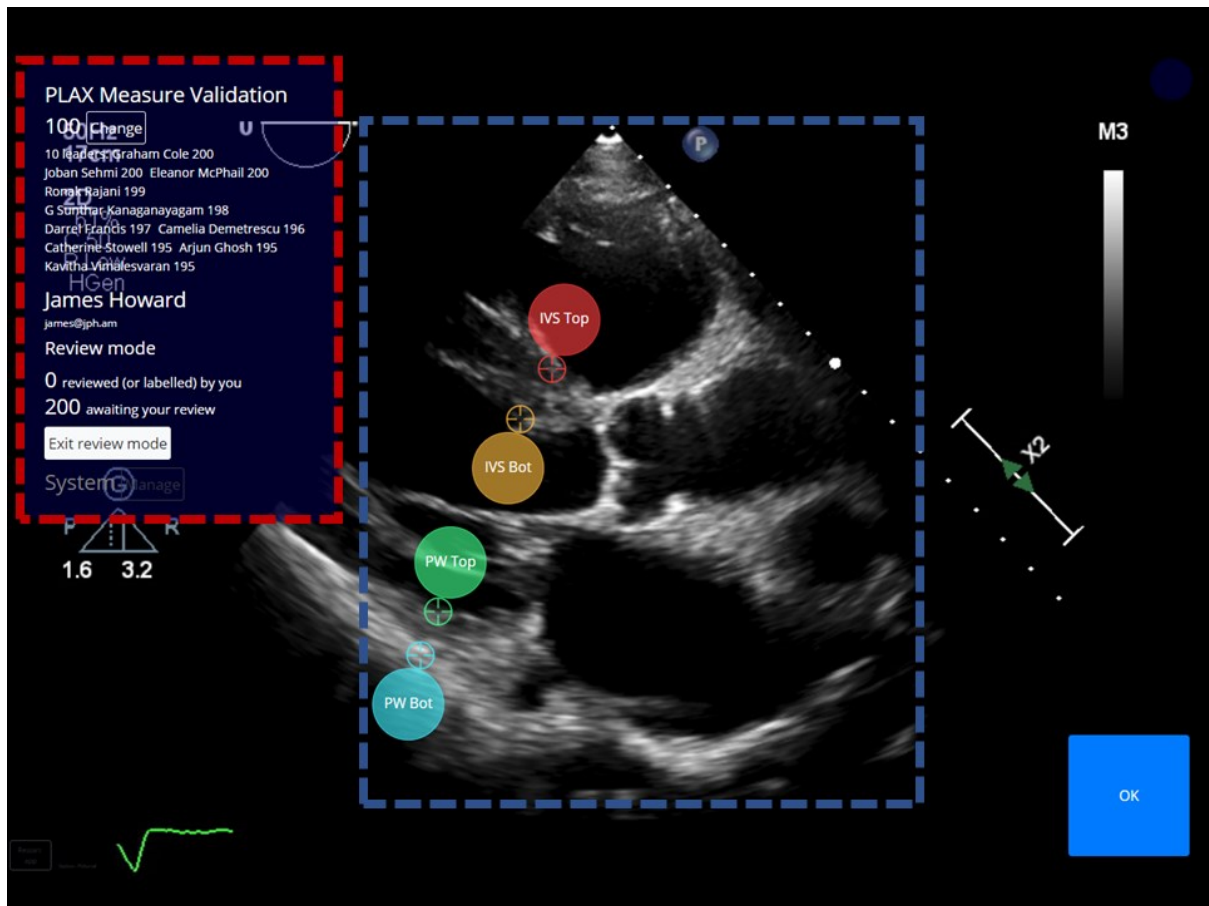


Figure 8-4 - Our an easy-to-use web-based interface to annotate medical images, which is currently in development. The system is divided into a labelled area (blue square) and an information area, showing that user's statistics, compared with those of other users (red square). Fiducial points on echocardiograms can be labelled either using either a mouse, or a touch screen interface such as that on a phone or tablet, by dragging the circles and their attached crosshairs.

8.4 Neural networks can point out diagnostically useful information

In this thesis I have used saliency mapping (described in section 2.10, page 90) to highlight aspects of data which are contributing most towards the neural networks' decisions. This is an established technique,⁵³ and has been successfully in settings where humans wish to ensure the neural network is using 'sensible' features of an image, rather than those which are unlikely to generalise well to new data (as in the wolf versus husky example in section 2.10).⁵⁴ I demonstrate this in section 6.5.2 where I show a temporal neural network appears to focus on the motion of the heart valve, which is a key feature humans use in identifying the echocardiogram view present. I also demonstrate in section 5.4.3 that a neural network which discriminates between the ECG responses to His bundle pacing appears to use similar aspects of the ECG as an expert human.

Another logical role for saliency mapping is using it to identify putative and previously unknown features that could assist human decision-making. I demonstrate this in section 4.5.3, where the saliency maps clearly identify unique circuit board components that discriminate between two otherwise-similar models of pacemaker.

However, in this thesis I may have identified a novel role for saliency mapping: using saliency maps to directly predict the likelihood of a neural network prediction being correct. In Figure 5-7 I demonstrate four representative saliency maps. The first three examples show correct predictions by the neural network, and the saliency maps demonstrate that features which are used by expert humans also appear to have greatly influenced the final decision of the network. However, the fourth example shows an incorrect prediction by the network, with a very different saliency map. In this example, the ECG features marked as salient are not those traditionally used by experts, and their utility are difficult to explain. I propose that in that incorrect example,

the neural network has failed to identify the correct features of the ECG on which it should make its decision. This begs the question: can saliency maps themselves be used to inform the confidence with which a network's decision should be relied on? Going forwards, I will investigate whether: (1) humans can identify cases where predictions by a neural network are more likely to be incorrect from the features highlighted in a saliency map (even when the humans are blinded to the neural network's decision; (2) whether a second neural network could similarly perform this task, trained to predict classification accuracy from a saliency maps alone.

8.5 AI will play an increasing role in medicine, with far reaching consequences

In this thesis I have demonstrated how just a single form of artificial intelligence (deep learning with convolutional neural networks) can achieve performances approaching (and sometimes exceeding) doctors in various tasks. Indeed, neural networks are now starting to become embedded in cardiologists' work flows, with automated biventricular segmentation of short axis images from cardiac magnetic resonance imaging scans now available in reporting software.¹¹⁵ Such solutions currently aim to improve the reporting process for doctors, replacing tedious tasks such as contour drawing.

However, there have been concerns that AI could end up not merely assisting but replacing doctors. Radiology, cardiac imaging, and pathology are specialties which involve highly skilled tasks, but ones where much of the clinical work requires little patient contact, and which many think AI could particularly excel at. Some have argued that many patients "would not realise" that their scan had been reported by a human or computer, and Vinod Khosla, co-founder of Sun Microsystems, argued as early as 2012 that "machines will replace 80 percent of doctors".¹¹⁶

In 2019, he went further, stating that “any radiologist who plans to practice in 10 years will be killing patients every day”, due to the inevitable superiority of AI.¹¹⁷

Others are more optimistic, arguing AI’s ability to decrease physicians’ workloads is not only attractive, but essential. As with echocardiographers,¹⁵ there is currently a shortfall of radiologists in the UK, and yet demand continues to increase by 4% per annum,¹¹⁸ equivalent to 50% growth per decade. AI may be our only solution to the inexorable rise in imaging. Curtis Langlotz, a radiologist at Stanford, famously stated, “AI won’t replace radiologists, but radiologists who use AI will replace radiologists who don’t”.

8.5.1 Why AI will not replace doctors

The AI solutions currently being developed, including those in this thesis, exist to assist doctors and not replace them. I strongly believe that in 10 years we will be far from being able to replace imaging doctors, let alone medical physicians, surgeons and general practitioners. I foresee three considerable barriers.

First, technical barriers. Whilst AI is beginning to approach human performance for many routine tasks, the difficulties of training effective AI models to deal with rare clinical diagnoses is currently far out of our grasp. Training a neural network to perform biventricular segmentation is possible because, for example, over 50,000 patients in the UK biobank have undergone cardiac imaging from which human ground truth labels can be derived, and a neural network trained. However, to develop a neural network that can accurately diagnose Chagas myocarditis would likely require a dataset larger than one that included every UK cardiac MRI exam depicting the diagnosis performed in the last decade.

Second, ‘explainability’. As I’ve addressed in this thesis, whilst AI solutions might be able to rival doctors’ performances in certain tasks, understanding *how* they have come to their decision is often difficult. Whilst in this thesis I have explored techniques such as saliency mapping to try

and reduce the opacity of some of the neural networks, they are not remotely adequate for difficult decision-making processes. Multi-disciplinary teams (MDTs) play an increasingly important role in doctors' jobs, and intelligent debate about optimal management strategies takes up a considerable proportion of these discussions. There is currently no way to replicate this. For example, even if a neural network were to recommend an ICD for a complex patient who does not meet the traditional criteria for a device, it would be of little help to the clinical team unless it could explain *why*.

Third, the human aspect. Relatively little progress has been made in replacing all but the simplest consultations with AI solutions. Whilst solutions such as Babylon Health's AI appointments have gained wide interest, their role is largely limited to primary care triaging, effectively utilising a flow chart algorithm. For the majority of consultations where a treatment or prescription is required, we are still fully reliant on highly-skilled and trained doctors and will be for the considerable future. The insurmountable barrier for AI will be that, given the choice of a skilled computer, and an equally skilled doctor, many of our patients would still prefer the latter.

Whilst eventually AI may indeed end up replacing *some* doctors, this will come after an era when it augments them. By then, we will have truly entered the age of AI, and medicine will certainly not be alone in this workforce revolution.

9 Conclusions

Machine learning is a revolution in computer science, where instead of programming a computer to perform a task, we program it in a way that it can learn to perform the task for itself.

Convolutional neural networks, inspired by the functioning of the mammalian optic cortex, are one possible approach. In this thesis, I have demonstrated the efficacy of these networks in X-rays, electrocardiograms, cardiac ultrasound and cardiac magnetic resonance imaging. In some settings I have shown that a convolutional neural network approach is able to achieve state-of-the-art performance beyond any available traditional algorithm. In others, I have used convolutional neural networks to solve problems for which no other method has proved successful. These networks will become increasingly commonplace as we enter the age of medical artificial intelligence, and I hope the work derived in this thesis will form the basis of my future research career.

10 Appendices

10.1 Publications arising from this PhD (2017-2020)

10.1.1 First author peer-reviewed publications

Howard JP, Fisher L, Shun-Shin MJ *et al.* “Cardiac Rhythm Device Identification Using Neural Networks.” **JACC Clin Electrophysiol.** 2019;5(5):576–586.

Howard JP, Cook CM, van de Hoef TP *et al.* “Artificial Intelligence for Aortic Pressure Waveform Analysis During Coronary Angiography – Machine Learning for Patient Safety.” **JACC Cardiovasc Interv.** 2019 Oct 28;12(20):2093-2101.

Howard JP, Tan J, Shun-Shin MJ *et al.* “Improving ultrasound video classification: an evaluation of novel deep learning methods in echocardiography.” **J Med Artif Intell.** 2020 Mar 25;3. pii: 4.

Howard & Arnold, Gopi AA, Chan CP *et al.* “Discriminating electrocardiographic responses to His-bundle pacing using machine learning” **Cardiovasc Digit Health J.** 2020 Jul-Aug; 1(1): 11–20.

Howard JP, Zaman S, Ragavan A *et al.* “Automated analysis and detection of abnormalities in transaxial anatomical cardiovascular magnetic resonance images: a proof of concept study with potential to optimize image acquisition” **Int J Cardiovasc Imaging.** 2020 Oct 29. doi: 10.1007/s10554-020-02050-w.

10.1.2 First author invited editorials

Howard JP, Murthy VL “A Song of Pressure and Flow, or There and Back Again.” **JACC Cardiovasc Interv.** 2018 Apr 23;11(8):754-756.

Howard JP “Renal denervation: The three stages of academic grief.” **Trends Cardiovasc Med.** 2019 Oct 22. pii: S1050-1738(19)30141-0.

10.1.3 Second author peer-reviewed publications

Ahmad Y, Howard JP, Arnold A *et al.* “Patent foramen ovale closure vs. medical therapy for cryptogenic stroke: a meta-analysis of randomized controlled trials.” **Eur Heart J.** 2018 May 7;39(18):1638-1649.

Al-Lamee R, Howard JP, Shun-Shin MJ *et al.* “Fractional Flow Reserve and Instantaneous Wave-Free Ratio as Predictors of the Placebo-Controlled Response to Percutaneous Coronary Intervention in Stable Single-Vessel Coronary Artery Disease.” **Circulation.** 2018 Oct 23;138(17):1780-1792.

Ahmad Y, Howard JP, Arnold *et al.* “Mortality after drug-eluting stents vs. coronary artery bypass grafting for left main coronary artery disease: a meta-analysis of randomized controlled trials” **Eur Heart J.** 2020 Mar 2. pii: ehaa135. [Epub ahead of print]

Ferreira-Martins J, Howard JP, Al-Khayatt B *et al.* “Outcomes of paroxysmal atrial fibrillation ablation studies are affected more by study design and patient mix than ablation technique.” **J Cardiovasc Electrophysiol.** 2018 Nov;29(11):1471-1479.

Arnold AD, Howard JP, Chiew K *et al.* “Right ventricular pacing for hypertrophic obstructive cardiomyopathy: meta-analysis and meta-regression of clinical trials.” **Eur Heart J Qual Care Clin Outcomes.** 2019 Oct 1;5(4):321-333.

Sau A, Howard JP, Al-Aidarous S *et al.* “Meta-analysis of randomized controlled trials of atrial fibrillation ablation with pulmonary vein isolation versus without” **JACC Clin Electrophysiol.** 2019 Aug;5(8):968-976

Cook CM, Howard JP, Ahmad Y *et al.* “How Do Fractional Flow Reserve, Whole-Cycle PdPa, and Instantaneous Wave-Free Ratio Correlate with Exercise Coronary Flow Velocity During Exercise-Induced Angina?” **Circ Cardiovasc Interv.** 2020 Apr;13(4):e008460.

Chacko L, Howard JP, Rajkumar C *et al.* “Effects of Percutaneous Coronary Intervention on Death and Myocardial Infarction Stratified by Stable and Unstable Coronary Artery Disease: A Meta-Analysis of Randomized Controlled Trials.” **Circ Cardiovasc Qual Outcomes.** 2020 Feb;13(2):e006363.

10.1.4 Other co-author publications

Al-Lamee R, Shun-Shin MJ, Howard JP *et al.* “Dobutamine Stress Echocardiography Ischemia as a Predictor of the Placebo-Controlled Efficacy of Percutaneous Coronary Intervention in Stable Coronary Artery Disease: The Stress Echocardiography-Stratified Analysis of ORBITA.” **Circulation.** 2019 Dec 10;140(24):1971-1980.

Cook CM, Ahmad Y, Howard JP *et al.* “Impact of Percutaneous Revascularization on Exercise Hemodynamics in Patients With Stable Coronary Disease.” **J Am Coll Cardiol.** 2018 Aug 28;72(9):970-983.

Arnold AD, Shun-Shin MJ, Keene D, Howard JP *et al.* “His Resynchronization Versus Biventricular Pacing in Patients With Heart Failure and Left Bundle Branch Block.” **J Am Coll Cardiol.** 2018 Dec 18;72(24):3112-3122.

Sen S, Ahmad Y, Dehbi HM, Howard JP *et al.* “Clinical Events After Deferral of LAD Revascularization Following Physiological Coronary Assessment” **J Am Coll Cardiol.** 2019 Feb 5;73(4):444-453.

Ahmad Y, Götzberg M, Cook C, Howard JP *et al.* “Coronary Hemodynamics in Patients With Severe Aortic Stenosis and Coronary Artery Disease Undergoing Transcatheter Aortic Valve Replacement: Implications for Clinical Indices of Coronary Stenosis Severity.” **JACC Cardiovasc Interv.** 2018 Oct 22;11(20):2019-2031.

Warisawa T, Cook CM, Howard JP *et al.* “Physiological Pattern of Disease Assessed by Pressure-Wire Pullback Has an Influence on Fractional Flow Reserve/Instantaneous Wave-Free Ratio Discordance.” **Circ Cardiovasc Interv.** 2019 May;12(5):e007494.

Keene D, Shun-Shin MJ, Arnold AD, Howard JP *et al.* “Quantification of electromechanical coupling to prevent inappropriate implantable cardioverter-defibrillator shocks” **JACC Clin Electrophysiol.** 2019 Jun;5(6):705-715.

Cook CM, Warisawa T, Howard JP *et al.* “Algorithmic versus expert human interpretation of instantaneous wave-free ratio coronary pressure-wire pull back data” **JACC Cardiovasc Interv.** 2019 Jul **22;12(14):1315-1324.**

Cook CM, Ahmad Y, Howard JP *et al.* “Association Between Physiological Stenosis Severity and Angina-Limited Exercise Time in Patients With Stable Coronary Artery Disease” **JAMA Cardiol.** 2019 Jun **1;4(6):569-574.**

Sau A, Al-Aidarous S, Howard JP *et al.* “Optimum lesion set and predictors of outcome in persistent atrial fibrillation ablation: a meta-regression analysis” **Europace.** 2019 Aug **1;21(8):1176-1184.**

Ahmad Y, Vendrik J, Eftekhari A, Howard JP *et al.* “Determining the Predominant Lesion in Patients With Severe Aortic Stenosis and Coronary Stenoses: A Multicenter Study Using Intracoronary Pressure and Flow” **Circ Cardiovasc Interv.** 2019 Dec;**12(12):e008263.**

Nowbar AN, Gitto M, Howard JP *et al.* “Mortality From Ischemic Heart Disease: Analysis of Data From the World Health Organization and Coronary Artery Disease Risk Factors From NCD Risk Factor Collaboration” **Circulation: Cardiovascular Quality and Outcomes** **12 (6), e005375**

10.2 Awards during this PhD

Society of Cardiovascular Magnetic Resonance Travel Scholarship (2020) – Awarded for my abstract “A deep learning strategy for rapid diagnosis from axial planning images: implications for optimal targeting of image acquisition”.

Best presentation – Imperial College Clinical Academic Training Conference (2019) – Awarded for my presentation, “Identifying pacemakers using AI in the emergency department”.

Supervisor of Best Project, Computational Medicine BSc, Imperial College (2018) – Awarded to the BSc student I co-supervised and who assisted me on the project “Cardiac Rhythm Device Identification Using Neural Networks”.

10.3 Software I have developed and released into the public domain during this PhD

10.3.1 BARD ECG segmentation & labelling software

For my work using neural networks to classify ECGs (Chapter 5), I created an open source program to allow exporting of ECGs from the proprietary 'BARD' electrophysiology application. The program allows labelling of ECG waveforms (Figure 5-2) and batch exporting (Figure 5-3) and is fully configurable via config files.

It is available at <https://github.com/jphdotam/BARD-labeller>

10.3.2 CoPhy

Current coronary physiology analysis packages typically require many steps, such as segmenting out SDY files and exporting them as 'TXT' files before analysis can even begin. I developed a free to use coronary physiology package that can calculate the common coronary physiology parameters including FFR, iFR, DFR, RFR, CFR, BSR, BMR and HMR. Parameters are given for full cycle, diastole, systole and the wave-free period (Figure 10-1). The software can read directly from SDY files, without requiring pre-processing.

It is available at <https://github.com/jphdotam/Cophy>

10.3.3 PyMinim – A simple python minimisation algorithm

I have developed a simple minimisation algorithm for 1:1 randomisation in trials. It allows a user to ‘intelligently’ randomise patients to arms in clinical trials to ensure baseline characteristics are optimally balanced.

It is available at <https://github.com/jphdotam/PyMinim>

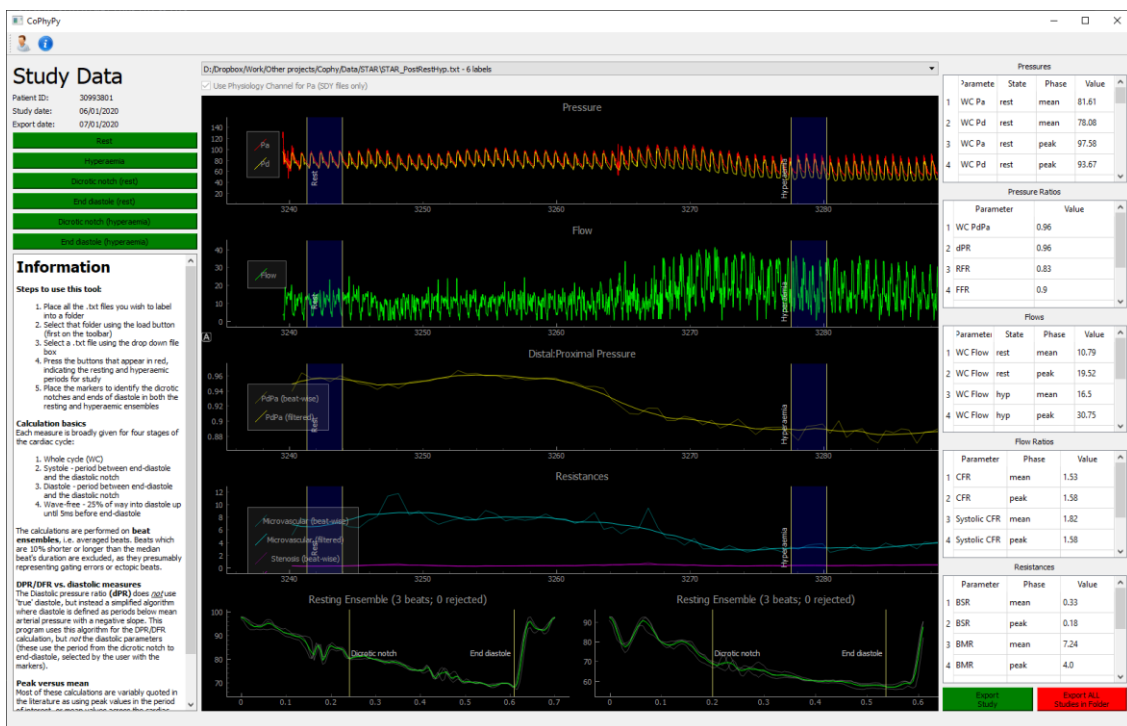


Figure 10-1 - A screenshot of the CoPhy coronary physiology suite.

11 References

1. De Marvao A, Dawes TJW, Howard JP, O'Regan DP. Artificial intelligence and the cardiologist: What you need to know for 2020 [Internet]. *Heart*. 2020 [cited 2020 Aug 20];106:399–400. Available from: <http://heart.bmj.com/>
2. Raatikainen MJP, Arnar DO, Merkely B, Nielsen JC, Hindricks G, Heidbuchel H, Camm J. A Decade of Information on the Use of Cardiac Implantable Electronic Devices and Interventional Electrophysiological Procedures in the European Society of Cardiology Countries: 2017 Report from the European Heart Rhythm Association. *Europace* [Internet]. 2017 [cited 2020 Aug 20];19:ii1–ii90. Available from: <https://pubmed.ncbi.nlm.nih.gov/28903470/>
3. Raatikainen MJP, Arnar DO, Zeppenfeld K, Merino JL, Levya F, Hindriks G, Kuck K-H. Statistics on the use of cardiac electronic devices and electrophysiological procedures in the European Society of Cardiology countries: 2014 report from the European Heart Rhythm Association. *Europace* [Internet]. 2015 [cited 2018 Jun 27];17:i1–i75. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/25616426>
4. Mond H, Proclemer A. The 11th World Survey of Cardiac Pacing and Implantable Cardioverter-Defibrillators: Calendar Year 2009-A World Society of Arrhythmia's Project. *Pacing Clin Electrophysiol* [Internet]. 2011 [cited 2018 Sep 4];34:1013–1027. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/21707667>
5. Jacob S, Shahzad MA, Maheshwari R, Panaich SS, Aravindhakshan R. Cardiac Rhythm Device Identification Algorithm using X-Rays: CaRDIA-X. *Heart Rhythm* [Internet]. 2011 [cited 2018 Jun 27];8:915–922. Available from: <https://www.sciencedirect.com/science/article/pii/S1547527111000142>

6. Kotsakou M, Kioumis I, Lazaridis G, Pitsiou G, Lampaki S, Papaiwannou A, Karavergou A, Tsakiridis K, Katsikogiannis N, Karapantzos I, Karapantzou C, Baka S, Mpoukovinas I, Karavasilis V, Rapti A, Trakada G, Zissimopoulos A, Zarogoulidis K, Zarogoulidis P. Pacemaker insertion. *Ann Transl Med* [Internet]. 2015 [cited 2020 Aug 20];3:42. Available from: [/pmc/articles/PMC4356861/?report=abstract](https://pmc/articles/PMC4356861/?report=abstract)

7. Brignole M, Auricchio A, Baron-Esquivias G, Bordachar P, Boriani G, Breithardt O-A, Cleland J, Deharo J-C, Delgado V, Elliott PM, Gorennek B, Israel CW, Leclercq C, Linde C, Mont L, Padeletti L, Sutton R, Vardas PE, Zamorano JL, Achenbach S, Baumgartner H, Bax JJ, Bueno H, Dean V, Deaton C, Erol C, Fagard R, Ferrari R, Hasdai D, Hoes AW, Kirchhof P, Knuuti J, Kolh P, Lancellotti P, Linhart A, Nihoyannopoulos P, Piepoli MF, Ponikowski P, Sirnes PA, Tamargo JL, Tendera M, Torbicki A, Wijns W, Windecker S, Kirchhof P, Blomstrom-Lundqvist C, Badano LP, Aliyev F, Bansch D, Baumgartner H, Bsata W, Buser P, Charron P, Daubert J-C, Dobreanu D, Faerestrand S, Hasdai D, Hoes AW, Le Heuzey J-Y, Mavrakis H, McDonagh T, Merino JL, Nawar MM, Nielsen JC, Pieske B, Poposka L, Ruschitzka F, Tendera M, Van Gelder IC, Wilson CM. 2013 ESC Guidelines on cardiac pacing and cardiac resynchronization therapy: The Task Force on cardiac pacing and resynchronization therapy of the European Society of Cardiology (ESC). Developed in collaboration with the European Heart Rhythm Association . *Europace*. 2013;

8. Gomes JAC, Damato AN, Akhtar M, Dhatt MS, Calon AH, Reddy CP, Moran HE. Ventricular septal motion and left ventricular dimensions during abnormal ventricular activation. *Am J Cardiol*. 1977;39:641–650.

9. Wilkoff BL, Cook JR, Epstein AE, Greene L, Hallstrom AP, Hsia H, Kutalek SP, Sharma A, Blatt B, Karas B, Kirchhoffer J, Warwick D, Duquette M, Provencher J, Redmond M, Herre JM, Bernstein R, Klevan LR, Barackman KD, Zumbuhl J, Chung MK, Jaeger FJ,

- Martin D, Natale A, Saliba WI, Schweikert RA, Niebauer MJ, Joseph Tchou P, Rozich R, Roelke M, Costeas CA, Rubenstein DG, Ruffo S, Kumar K, McCarthy E, Pastore V, Wathen MS, Rottman J, Anderson M, Lee JT, Murray KT, Roden DM, Connors N, Saunders S, O'Neill GP, Skadsen A, Allen S, Vierra E, Greer S, Neuhauser J, Myers P, Lee C, Moore T, Klein RC, Freedman RA, Wadsworth G, Dailey SM, Neal Kay G, Plumb VJ, Bubien RS, Kay LW, Nasser CM, Slabaugh JE, Leman R, Lake JL, Clark J, Clarke E, Finklea L, Love JC, Carpenter CM, Corsello A, Cutler JE, BosworthFarrell S, Michaud G, Buxton AE, Ellison KE, Christian F, Kirk MM, Corcoran PL, Rothbart S, Sauberman RB, McCarthy J, Page ME, Steinberg JS, Ehlert F, Herweg B, Vloka M, Malinay A, Menchavez E, Rome M, Marks K, Swarens A, Hernandez M, Marinchak R, Esberg D, Finkle J, Harper G, Kowey PR, et al. Dual-chamber pacing-or ventricular backup pacing in patients with an implantable defibrillator: The Dual Chamber and VVI Implantable Defibrillator (DAVID) Trial. *J Am Med Assoc*. 2002;288:3115–3123.
10. Deshmukh P, Casavant DA, Romanyshyn M, Anderson K. Permanent, Direct His-Bundle Pacing. *Circulation* [Internet]. 2000 [cited 2020 Jun 12];101:869–877. Available from: <https://www.ahajournals.org/doi/10.1161/01.CIR.101.8.869>
 11. Arnold AD, Shun-Shin MJ, Keene D, Howard JP, Sohaib SMA, Wright IJ, Cole GD, Qureshi NA, Lefroy DC, Koa-Wing M, Linton NWF, Lim PB, Peters NS, Davies DW, Muthumala A, Tanner M, Ellenbogen KA, Kanagaratnam P, Francis DP, Whinnett ZI. His Resynchronization Versus Biventricular Pacing in Patients With Heart Failure and Left Bundle Branch Block. *J Am Coll Cardiol* [Internet]. 2018 [cited 2019 Aug 27];72:3112–3122. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/30545450>
 12. Burri H, Jastrzebski M, Vijayaraman P. Electrocardiographic Analysis for His Bundle Pacing at Implantation and Follow-Up [Internet]. *JACC Clin. Electrophysiol*. 2020 [cited 2020 Aug 17];6:883–900. Available from: <https://pubmed.ncbi.nlm.nih.gov/32703577/>

13. Howard JP, Cook CM, van de Hoef TP, Meuwissen M, de Waard GA, van Lavieren MA, Echavarria-Pinto M, Danad I, Piek JJ, Götberg M, Al-Lamee RK, Sen S, Nijjer SS, Seligman H, van Royen N, Knaapen P, Escaned J, Francis DP, Petraco R, Davies JE. Artificial Intelligence for Aortic Pressure Waveform Analysis During Coronary Angiography. *JACC Cardiovasc Interv.* 2019;
14. Virnig BA, Shippee ND, O'Donnell B, Zeglin J, Parashuram S. Trends in the use of echocardiography, 2007 to 2011 [Internet]. Agency for Healthcare Research and Quality (US); 2011 [cited 2019 May 14]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/24967475>
15. British Society of Echocardiography. Cardiac Physiology Workforce Options Analysis [Internet]. 2016 [cited 2019 May 14]. Available from: https://www.bsecho.org/media/195270/future_workforce_options_analysis_01.11.2016_final.pdf
16. Nowbar AN, Cole GD, Francis DP. The changing relative contribution of operator salary and hardware costs to echocardiography — A historical perspective. *Int J Cardiol* [Internet]. 2013 [cited 2018 Oct 26];168:1641. Available from: <https://www.sciencedirect.com/science/article/pii/S0167527313003264?via%3Dihub>
17. Curiale AH, Colavecchia FD, Kaluza P, Isoardi RA, Mato G. Automatic Myocardial Segmentation by Using A Deep Learning Network in Cardiac MRI. 2017 [cited 2019 Jul 10]; Available from: <http://arxiv.org/abs/1708.07452>
18. Bai W, Oktay O, Sinclair M, Suzuki H, Rajchl M, Tarroni G, Glocker B, King A, Matthews PM, Rueckert D. Semi-supervised Learning for Network-Based Cardiac MR Image Segmentation [Internet]. Springer, Cham; 2017 [cited 2019 Jul 10]. p. 253–260. Available from: http://link.springer.com/10.1007/978-3-319-66185-8_29

19. Patravali J, Jain S, Chilamkurthy S. 2D-3D Fully Convolutional Neural Networks for Cardiac MR Segmentation. 2017 [cited 2019 Jul 10];Available from: <http://arxiv.org/abs/1707.09813>
20. Payer C, Štern D, Bischof H, Urschler M. Multi-label Whole Heart Segmentation Using CNNs and Anatomical Label Configurations [Internet]. Springer, Cham; 2018 [cited 2019 Jul 10]. p. 190–198.Available from: http://link.springer.com/10.1007/978-3-319-75541-0_20
21. Cano-Espinosa C, González G, Washko GR, Cazorla M, Estépar RSJ. Automated Agatston Score Computation in non-ECG Gated CT Scans Using Deep Learning. *Proc SPIE--the Int Soc Opt Eng* [Internet]. 2018 [cited 2019 Jul 10];10574. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/30122801>
22. López-Linares K, Kabongo L, Lete N, Maclair G, Ceresa M, García-Familiar A, Macía I, González Ballester MÁ. DCNN-Based Automatic Segmentation and Quantification of Aortic Thrombus Volume: Influence of the Training Approach [Internet]. Springer, Cham; 2017 [cited 2019 Jul 10]. p. 29–38.Available from: http://link.springer.com/10.1007/978-3-319-67534-3_4
23. Liu H, Feng J, Feng Z, Lu J, Zhou J. Left Atrium Segmentation in CT Volumes with Fully Convolutional Networks [Internet]. Springer, Cham; 2017 [cited 2019 Jul 10]. p. 39–46.Available from: http://link.springer.com/10.1007/978-3-319-67558-9_5
24. Zreik M, Lessmann N, van Hamersvelt RW, Wolterink JM, Voskuil M, Viergever MA, Leiner T, Išgum I. Deep learning analysis of the myocardium in coronary CT angiography for identification of patients with functionally significant coronary artery stenosis. *Med Image Anal* [Internet]. 2018 [cited 2019 Jul 10];44:72–85. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/29197253>

25. Madani A, Arnaout R, Mofrad M, Arnaout R. Fast and accurate view classification of echocardiograms using deep learning. *npj Digit Med* [Internet]. 2018 [cited 2018 Jul 2];1. Available from: <https://www.nature.com/articles/s41746-017-0013-1.pdf?origin=ppub>
26. Zhang J, Gajjala S, Agrawal P, Tison GH, Hallock LA, Beussink-Nelson L, Lassen MH, Fan E, Aras MA, Jordan C, Fleischmann KE, Melisko M, Qasim A, Shah SJ, Bajcsy R, Deo RC. Fully Automated Echocardiogram Interpretation in Clinical Practice. *Circulation* [Internet]. 2018 [cited 2019 Apr 1];138:1623–1635. Available from: <https://www.ahajournals.org/doi/10.1161/CIRCULATIONAHA.118.034338>
27. Petersen SE, Aung N, Sanghvi MM, Zemrak F, Fung K, Paiva JM, Francis JM, Khanji MY, Lukaschuk E, Lee AM, Carapella V, Kim YJ, Leeson P, Piechnik SK, Neubauer S. Reference ranges for cardiac structure and function using cardiovascular magnetic resonance (CMR) in Caucasians from the UK Biobank population cohort. *J Cardiovasc Magn Reson*. 2017;19.
28. Kramer CM, Barkhausen J, Flamm SD, Kim RJ, Nagel E. Standardized cardiovascular magnetic resonance (CMR) protocols 2013 update. *J Cardiovasc Magn Reson* [Internet]. 2013 [cited 2019 Oct 9];15:91. Available from: <https://jcmr-online.biomedcentral.com/articles/10.1186/1532-429X-15-91>
29. Rodrigues JCL, Lyen SM, Loughborough W, Amadu AM, Baritussio A, Dastidar AG, Manghat NE, Bucciarelli-Ducci C. Extra-cardiac findings in cardiovascular magnetic resonance: What the imaging cardiologist needs to know [Internet]. *J. Cardiovasc. Magn. Reson*. 2016 [cited 2020 Apr 9];18:26. Available from: <http://jcmr-online.biomedcentral.com/articles/10.1186/s12968-016-0246-1>
30. Lecun Y, Bottou L, Bengio Y, Ha P. LeNet. *Proc IEEE*. 1998;
31. Xia Y, Wulan N, Wang K, Zhang H. Detecting atrial fibrillation by deep convolutional

- neural networks. *Comput Biol Med* [Internet]. 2018 [cited 2019 Jul 10];93:84–92. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/29291535>
32. Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, Ding D, Bagul A, Ball RL, Langlotz C, Shpanskaya K, Lungren MP, Ng AY. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. 2017 [cited 2018 Jun 27];Available from: <https://arxiv.org/pdf/1711.05225.pdf>
 33. Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. Learning Spatiotemporal Features with 3D Convolutional Networks [Internet]. [cited 2019 Apr 24]. Available from: <https://arxiv.org/pdf/1412.0767.pdf>
 34. Nes BM, Janszky I, Wisløff U, Støylen A, Karlsen T. Age-predicted maximal heart rate in healthy subjects: The HUNT Fitness Study. *Scand J Med Sci Sport*. 2013;
 35. Ahmad T, Lund LH, Rao P, Ghosh R, Warier P, Vaccaro B, Dahlström U, O'Connor CM, Michael Felker G, Desai NR. Machine learning methods improve prognostication, identify clinically distinct phenotypes, and detect heterogeneity in response to therapy in a large cohort of heart failure patients. *J Am Heart Assoc* [Internet]. 2018 [cited 2020 Aug 17];7. Available from: <https://www.ahajournals.org/doi/10.1161/JAHA.117.008081>
 36. Ruder S. An overview of gradient descent optimization algorithms. 2016 [cited 2020 Aug 20];Available from: <http://arxiv.org/abs/1609.04747>
 37. Dally B. Efficient methods and hardware for deep learning. 2017.
 38. Nair V, Hinton GE. Rectified linear units improve Restricted Boltzmann machines. In: ICML 2010 - Proceedings, 27th International Conference on Machine Learning. 2010.
 39. Huang D, Yamauchi K, Inden Y, Yang J, Jiang Z, Ida H, Katsuyama K, Wang K, Kato K, Kato H. Use of an artificial neural network to localize accessory pathways of Wolff–

- Parkinson–White syndrome with 12-lead electrocardiogram. *Med Inform Internet Med* [Internet]. 2005 [cited 2019 Jul 10];30:277–286. Available from: <http://www.tandfonline.com/doi/full/10.1080/14639230500367670>
40. Zheng Y, Liu D, Georgescu B, Nguyen H, Comaniciu D. 3D Deep Learning for Efficient and Robust Landmark Detection in Volumetric Data [Internet]. Springer, Cham; 2015 [cited 2019 Jul 10]. p. 565–572. Available from: http://link.springer.com/10.1007/978-3-319-24553-9_69
 41. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, Thrun S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature* [Internet]. 2017 [cited 2019 May 7];542:115–118. Available from: <http://www.nature.com/articles/nature21056>
 42. Hubel DH, Wiesel TN. Shape and arrangement of columns in cat's striate cortex. *J Physiol* [Internet]. 1963 [cited 2019 Jul 10];165:559–568. Available from: <http://doi.wiley.com/10.1113/jphysiol.1963.sp007079>
 43. Schiltz C, Sorger B, Caldara R, Ahmed F, Mayer E, Goebel R, Rossion B. Impaired Face Discrimination in Acquired Prosopagnosia Is Associated with Abnormal Response to Individual Faces in the Right Middle Fusiform Gyrus. *Cereb Cortex* [Internet]. 2006 [cited 2019 Jul 10];16:574–586. Available from: <http://academic.oup.com/cercor/article/16/4/574/382305/Impaired-Face-Discrimination-in-Acquired>
 44. HUBEL DH, WIESEL TN. RECEPTIVE FIELDS OF CELLS IN STRIATE CORTEX OF VERY YOUNG, VISUALLY INEXPERIENCED KITTENS. *J Neurophysiol* [Internet]. 1963 [cited 2019 Jul 10];26:994–1002. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/14084171>
 45. Poh M-Z, Poh YC, Chan P-H, Wong C-K, Pun L, Leung WW-C, Wong Y-F, Wong MM-

- Y, Chu DW-S, Siu C-W. Diagnostic assessment of a deep learning system for detecting atrial fibrillation in pulse waveforms. *Heart* [Internet]. 2018 [cited 2018 Jun 27];heartjnl-2018-313147. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/29853485>
46. He K, Gkioxari G, Dollar P, Girshick R. Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. 2017.
 47. Badrinarayanan V, Handa A, Cipolla R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling. 2015 [cited 2017 May 18];Available from: <http://arxiv.org/abs/1505.07293>
 48. Shorten C, Khoshgoftaar TM. A survey on Image Data Augmentation for Deep Learning. *J Big Data*. 2019;
 49. Pratt LY, Hansont SJ. Comparing Biases for Minimal Network Construction with Back-Propagation. *NIPS Proc* [Internet]. 1989 [cited 2020 Feb 18];177–185. Available from: <https://papers.nips.cc/paper/156-comparing-biases-for-minimal-network-construction-with-back-propagation>
 50. Srivastava N, Hinton G, Krizhevsky A, Salakhutdinov R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. 2014.
 51. Benitez JM, Castro JL, Requena I. Are artificial neural networks black boxes? *IEEE Trans Neural Networks* [Internet]. 1997 [cited 2018 Aug 23];8:1156–1164. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/18255717>
 52. Schwartz-Ziv R, Tishby N. Opening the black box of Deep Neural Networks via Information [Internet]. 2017 [cited 2018 Aug 23]. Available from: <https://arxiv.org/pdf/1703.00810.pdf>
 53. Simonyan K, Vedaldi A, Zisserman A. Deep Inside Convolutional Networks: Visualising

- Image Classification Models and Saliency Maps [Internet]. [cited 2018 Aug 20]. Available from: <http://code.google.com/p/cuda-convnet/>
54. Ribeiro MT, Singh S, Guestrin C. “Why should i trust you?” Explaining the predictions of any classifier [Internet]. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, New York, USA: Association for Computing Machinery; 2016 [cited 2020 Apr 7]. p. 1135–1144. Available from: <http://dl.acm.org/citation.cfm?doid=2939672.2939778>
 55. Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks [Internet]. [cited 2019 Jul 10]. Available from: <http://code.google.com/p/cuda-convnet/>
 56. UK Biobank. UK BioBank [Internet]. [cited 2019 Jul 10]; Available from: <https://www.ukbiobank.ac.uk/>
 57. Google. Google AI Blog: DeepDream - a code example for visualizing Neural Networks [Internet]. [cited 2020 Aug 17]; Available from: <https://ai.googleblog.com/2015/07/deepdream-code-example-for-visualizing.html>
 58. Stultz CM. The Advent of Clinically Useful Deep Learning. *JACC Clin Electrophysiol* [Internet]. 2019 [cited 2019 Jul 10];5:587–589. Available from: <https://linkinghub.elsevier.com/retrieve/pii/S2405500X19301690>
 59. U.S. Food & Drug Administration. Software as a Medical Device (SaMD) | FDA [Internet]. [cited 2020 Aug 17]; Available from: <https://www.fda.gov/medical-devices/digital-health/software-medical-device-samd>
 60. “Python Software Foundation.” The Python Programming Language [Internet]. [cited 2017 Jul 9]; Available from: <https://www.python.org/>

61. Towards Data Science. What is the best programming language for Machine Learning? [Internet]. 2017 [cited 2020 Mar 27];Available from: <https://towardsdatascience.com/what-is-the-best-programming-language-for-machine-learning-a745c156d6b7>
62. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mane D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viegas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2016 [cited 2017 Jul 9];Available from: <http://arxiv.org/abs/1603.04467>
63. Chollet F. Keras: The Python Deep Learning library [Internet]. 2015;Available from: www.keras.io
64. Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A. Automatic differentiation in PyTorch [Internet]. 2017 [cited 2019 Sep 4];Available from: <https://openreview.net/forum?id=BJJsrmfCZ>
65. R Core Team. R: A Language and Environment for Statistical Computing [Internet]. 2014;Available from: <http://www.r-project.org/>
66. Wickham H. ggplot2: elegant graphics for data analysis [Internet]. Springer New York; 2009. Available from: <http://had.co.nz/ggplot2/book>
67. Howard JP, Fisher L, Shun-Shin MJ, Keene D, Arnold AD, Ahmad Y, Cook CM, Moon JC, Manisty CH, Whinnett ZI, Cole GD, Rueckert D, Francis DP. Cardiac Rhythm Device Identification Using Neural Networks. *JACC Clin Electrophysiol* [Internet]. 2019 [cited 2019 May 28];5:576–586. Available from:

<http://www.ncbi.nlm.nih.gov/pubmed/31122379>

68. Mazurowski MA, Habas PA, Zurada JM, Lo JY, Baker JA, Tourassi GD. Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks* [Internet]. 2008 [cited 2018 Jun 27];21:427–436. Available from:
<https://www.sciencedirect.com/science/article/pii/S0893608007002407>
69. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2015 [cited 2018 Jun 27];Available from:
<https://arxiv.org/pdf/1409.1556.pdf>
70. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the Inception Architecture for Computer Vision. [cited 2018 Jun 27];Available from:
<https://arxiv.org/pdf/1512.00567.pdf>
71. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. [cited 2018 Jun 27];Available from: <https://arxiv.org/pdf/1512.03385.pdf>
72. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely Connected Convolutional Networks. [cited 2018 Jun 27];Available from: <https://arxiv.org/pdf/1608.06993.pdf>
73. Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions [Internet]. [cited 2018 Aug 20]. Available from:
http://openaccess.thecvf.com/content_cvpr_2017/papers/Chollet_Xception_Deep_Learning_CVPR_2017_paper.pdf
74. Zeiler MD. ADADELTA: An Adaptive Learning Rate Method. 2012 [cited 2018 Jun 27];Available from: <http://arxiv.org/abs/1212.5701>
75. Kotikalapudi R. Keras-vis - Keras Visualization Toolkit [Internet]. 2018 [cited 2018 Jun

28];Available from: <https://raghakot.github.io/keras-vis/>

76. Bansal R, Raj G, Choudhury T. Blur image detection using Laplacian operator and Open-CV [Internet]. In: 2016 International Conference System Modeling & Advancement in Research Trends (SMART). IEEE; 2016 [cited 2019 Jan 1]. p. 63–67. Available from: <http://ieeexplore.ieee.org/document/7894491/>
77. Enache B. Bogdan Enache on Twitter [Internet]. Twitter. [cited 2020 Aug 20]; Available from: <https://twitter.com/bogdienache/status/1111666593954709504>
78. Arnold AD, Howard JP, Gopi AA, Chan CP, Ali N, Keene D, Shun-Shin MJ, Ahmad Y, Wright IJ, Ng FS, Linton NWF, Kanagaratnam P, Peters NS, Rueckert D, Francis DP, Whinnett ZI. Discriminating electrocardiographic responses to His-bundle pacing using machine learning. *Cardiovasc Digit Heal J*. 2020;
79. Moss AJ, Hall WJ, Cannom DS, Klein H, Brown MW, Daubert JP, Estes NAM, Foster E, Greenberg H, Higgins SL, Pfeffer MA, Solomon SD, Wilber D, Zareba W, Desai P, Wiggins S, Greer G, Beau S, Curnis A, Katz A, Cook J, McPherson C, Rozmus G, Switzer D, Stone J, Ludmer P, Colavita P, Tomassoni G, Crevey B, Nair G, Saliba W, Corbisiero R, Gilliam F, Hranitzky P, Rashtian M, Giudici M, Thomsen P, Clyne C, Pena E, Lessmeier T, Schuger C, Vogt J, Kacet S, Almendral J, Quesada A, Kautzner J, Padeletti L, Delnoy P, Goel S, Berger R, Pitschner H, Martin D, Kfoury A, Klein S, Levin V, Schalij M, Chow T, Chung E, Greenberg Y, Lemke B, Singh J, Rea R, Gold M, Guttigoli A, Adler A, Singer I, Shinn T, Guarnieri T, Casey C, Naccarelli G, Gornick C, Thibault B, Ackerman S, Turk K, Hunter N, Jentzer J, Bartlett T, Glascock D, Tamirisa K, Goldberger J, Coman J, Sandler D, Malik R, Nair L, O'Neill P, Sharma A, Brodine W, Kargul W, Porter M, Merkely B, Onufer J, Eldar M, Gottipaty V, Pires L, Wilson D, Arshad A, Fischer A, Mollerus M, et al. Cardiac-resynchronization therapy for the prevention of heart-failure events. *N Engl J Med*. 2009;361:1329–1338.

80. Curtis AB, Worley SJ, Adamson PB, Chung ES, Niazi I, Sherfese L, Shinn T, St. John Sutton M. Biventricular Pacing for Atrioventricular Block and Systolic Dysfunction. *N Engl J Med* [Internet]. 2013 [cited 2020 Jun 12];368:1585–1593. Available from: <http://www.nejm.org/doi/10.1056/NEJMoa1210356>
81. Huang W, Su L, Wu S, Xu L, Xiao F, Zhou X, Mao G, Vijayaraman P, Ellenbogen KA. Long-term outcomes of His bundle pacing in patients with heart failure with left bundle branch block. *Heart*. 2019;105:137–143.
82. Sharma PS, Naperkowski A, Bauch TD, Chan JYS, Arnold AD, Whinnett ZI, Ellenbogen KA, Vijayaraman P. Permanent His Bundle Pacing for Cardiac Resynchronization Therapy in Patients With Heart Failure and Right Bundle Branch Block. *Circ Arrhythm Electrophysiol*. 2018;11:e006613.
83. Qian Z, Zou F, Wang Y, Qiu Y, Chen X, Jiang H, Hou X. Permanent His bundle pacing in heart failure patients: A systematic review and meta-analysis. *PACE - Pacing Clin. Electrophysiol*. 2019;42:139–145.
84. Macfarlane PW, Lawrie TDV. Automated e.c.g. interpretation in the National Health Service. *Proc R Soc London Ser B Biol Sci* [Internet]. 1973 [cited 2020 Jun 14];184:455–471. Available from: <https://royalsocietypublishing.org/doi/10.1098/rspb.1973.0064>
85. Graves A, Mohamed A, Hinton G. Speech Recognition with Deep Recurrent Neural Networks. *ICASSP, IEEE Int Conf Acoust Speech Signal Process - Proc* [Internet]. 2013 [cited 2020 Jun 14];6645–6649. Available from: <http://arxiv.org/abs/1303.5778>
86. Lipton ZC, Berkowitz J, Elkan C. A Critical Review of Recurrent Neural Networks for Sequence Learning. 2015 [cited 2020 Jun 14];Available from: <http://arxiv.org/abs/1506.00019>
87. Salloum R, Kuo CCJ. ECG-based biometrics using recurrent neural networks. In:

- ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings. Institute of Electrical and Electronics Engineers Inc.; 2017. p. 2062–2066.
88. Cheng M, Sori WJ, Jiang F, Khan A, Liu S. Recurrent Neural Network Based Classification of ECG Signal Features for Obstruction of Sleep Apnea Detection. In: Proceedings - 2017 IEEE International Conference on Computational Science and Engineering and IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, CSE and EUC 2017. Institute of Electrical and Electronics Engineers Inc.; 2017. p. 199–202.
 89. Hannun AY, Rajpurkar P, Haghpanahi M, Tison GH, Bourn C, Turakhia MP, Ng AY. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nat Med* [Internet]. 2019 [cited 2019 Apr 9];25:65–69. Available from: <http://www.nature.com/articles/s41591-018-0268-3>
 90. Zagoruyko S, Komodakis N. Wide Residual Networks. *Br Mach Vis Conf 2016, BMVC 2016* [Internet]. 2016 [cited 2020 Jun 29];2016-September:87.1-87.12. Available from: <http://arxiv.org/abs/1605.07146>
 91. Xie S, Girshick R, Dollár P, Tu Z, He K, San Diego U. Aggregated Residual Transformations for Deep Neural Networks [Internet]. [cited 2020 Jun 29]. Available from: <https://github.com/facebookresearch/ResNeXt>
 92. Singh S, Pandey SK, Pawar U, Janghel RR. Classification of ECG Arrhythmia using Recurrent Neural Networks. In: *Procedia Computer Science*. Elsevier B.V.; 2018. p. 1290–1297.
 93. Brito Teixeira CM, Pereira TA, Lebreiro AM, Carvalho SA. Accuracy of the electrocardiogram in localizing the accessory pathway in patients with Wolff-Parkinson-White pattern. *Arq Bras Cardiol* [Internet]. 2016 [cited 2020 Jul 1];107:331–338. Available

from: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0066-782X2016004300331&lng=en&nrm=iso&tlng=en

94. Howard JP, Tan J, Shun-Shin MJ, Mahdi D, Nowbar AN, Arnold AD, Ahmad Y, McCartney P, Zolgharni M, Linton NWF, Sutaria N, Rana B, Mayet J, Rueckert D, Cole GD, Francis DP. Improving ultrasound video classification: an evaluation of novel deep learning methods in echocardiography. *J Med Artif Intell.* 2020;3:4–4.
95. Zhao Q. py-denseflow [Internet]. GitHub. 2017;Available from: <https://github.com/qijiezhao/py-denseflow/>
96. Donahue J, Hendricks LA, Rohrbach M, Venugopalan S, Guadarrama S, Saenko K, Darrell T. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. 2014 [cited 2019 Jan 15];Available from: <http://arxiv.org/abs/1411.4389>
97. Lin M, Chen Q, Yan S. Network In Network [Internet]. [cited 2019 Apr 24]. Available from: <https://arxiv.org/pdf/1312.4400.pdf>
98. Carreira J, Zisserman A, Com Z, Deepmind †. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset [Internet]. [cited 2019 Apr 21]. Available from: <https://arxiv.org/pdf/1705.07750.pdf>
99. Deepmind. I3D models trained on Kinetics. GitHub. 2017;
100. Feichtenhofer C, Pinz A, Zisserman A. Convolutional Two-Stream Network Fusion for Video Action Recognition. 2016 [cited 2019 Apr 21];Available from: <http://arxiv.org/abs/1604.06573>
101. Howard JP, Zaman S, Ragavan A, Hall K, Leonard G, Sutanto S, Ramadoss V, Razvi Y, Linton NF, Bharath A, Shun-Shin M, Rueckert D, Francis D, Cole G. Automated analysis and detection of abnormalities in transaxial anatomical cardiovascular magnetic resonance

- images: a proof of concept study with potential to optimize image acquisition. *Int J Cardiovasc Imaging* [Internet]. 2020 [cited 2020 Nov 11];1–10. Available from: <https://doi.org/10.1007/s10554-020-02050-w>
102. Bhuva AN, Bai W, Lau C, Davies RH, Ye Y, Bulluck H, McAlindon E, Culotta V, Swoboda PP, Captur G, Treibel TA, Augusto JB, Knott KD, Seraphim A, Cole GD, Petersen SE, Edwards NC, Greenwood JP, Bucciarelli-Ducci C, Hughes AD, Rueckert D, Moon JC, Manisty CH. A Multicenter, Scan-Rescan, Human and Machine Learning CMR Study to Test Generalizability and Precision in Imaging Biomarker Analysis. *Circ Cardiovasc Imaging* [Internet]. 2019 [cited 2019 Oct 9];12:e009214. Available from: <https://www.ahajournals.org/doi/10.1161/CIRCIMAGING.119.009214>
 103. Ruijsink B, Puyol-Antón E, Oksuz I, Sinclair M, Bai W, Schnabel JA, Razavi R, King AP. Fully Automated, Quality-Controlled Cardiac Analysis From CMR. *JACC Cardiovasc Imaging*. 2019;
 104. Bai W, Sinclair M, Tarroni G, Oktay O, Rajchl M, Vaillant G, Lee AM, Aung N, Lukaschuk E, Sanghvi MM, Zemrak F, Fung K, Paiva JM, Carapella V, Kim YJ, Suzuki H, Kainz B, Matthews PM, Petersen SE, Piechnik SK, Neubauer S, Glocker B, Rueckert D. Automated cardiovascular magnetic resonance image analysis with fully convolutional networks. *J Cardiovasc Magn Reson* [Internet]. 2018 [cited 2020 Jan 21];20:65. Available from: <https://jcmr-online.biomedcentral.com/articles/10.1186/s12968-018-0471-x>
 105. Sun K, Zhao Y, Jiang B, Cheng T, Xiao B, Liu D, Mu Y, Wang X, Liu W, Wang J. High-Resolution Representations for Labeling Pixels and Regions. 2019 [cited 2019 Aug 30];Available from: <http://arxiv.org/abs/1904.04514>
 106. Liu L, Jiang H, He P, Chen W, Liu X, Gao J, Han J. On the Variance of the Adaptive Learning Rate and Beyond. 2019 [cited 2019 Oct 10];Available from:

<http://arxiv.org/abs/1908.03265>

107. Zhang MR, Lucas J, Hinton G, Ba J. Lookahead Optimizer: k steps forward, 1 step back. 2019 [cited 2019 Oct 10]; Available from: <http://arxiv.org/abs/1907.08610>
108. Kawel-Boehm N, Maceira A, Valsangiacomo-Buechel ER, Vogel-Claussen J, Turkbey EB, Williams R, Plein S, Tee M, Eng J, Bluemke DA. Normal values for cardiovascular magnetic resonance in adults and children. *J Cardiovasc Magn Reson* [Internet]. 2015 [cited 2019 Nov 25];17:29. Available from: <http://jcmr-online.com/content/17/1/29>
109. Karamitsos TD, Piechnik SK, Bannypersad SM, Fontana M, Ntusi NB, Ferreira VM, Whelan CJ, Myerson SG, Robson MD, Hawkins PN, Neubauer S, Moon JC. Noncontrast T1 mapping for the diagnosis of cardiac amyloidosis. *JACC Cardiovasc Imaging*. 2013;6:488–497.
110. Stuppel A, Singerman D, Celi LA. The reproducibility crisis in the age of digital medicine. *npj Digit Med*. 2019;2.
111. Collins GS, Altman DG. An independent and external validation of QRISK2 cardiovascular disease risk score: A prospective open cohort study. *BMJ*. 2010;340:1231.
112. Arruda MS, McClelland JH, Wang X, Beckman KJ, Widman LE, Gonzalez MD, Nakagawa H, Lazzara R, Jackman WM. Development and validation of an ECG algorithm for identifying accessory pathway ablation site in Wolff-Parkinson-White syndrome. *J Cardiovasc Electrophysiol* [Internet]. 1998 [cited 2019 Jan 8];9:2–12. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/9475572>
113. Deng J, Dong W, Socher R, Li L-J, Kai Li, Li Fei-Fei. ImageNet: A large-scale hierarchical image database. Institute of Electrical and Electronics Engineers (IEEE); 2010. p. 248–255.

114. Hu J, Shen L, Albanie S, Sun G, Wu E. Squeeze-and-Excitation Networks. *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit* [Internet]. 2017 [cited 2020 Jul 6];7132–7141. Available from: <http://arxiv.org/abs/1709.01507>
115. Circle Cardiovascular Imaging. Cardiac MRI and CT Software – Circle Cardiovascular Imaging - Deep Learning [Internet]. [cited 2020 Nov 29];Available from: <https://www.circlecvi.com/cvi42/cardiac-mri/deep-learning/>
116. Vinod Khosla: Machines will replace 80 percent of doctors | WIRED UK [Internet]. [cited 2020 Nov 29];Available from: <https://www.wired.co.uk/article/doctors-replaced-with-machines>
117. Why Vinod Khosla thinks radiologists still practicing in 10 years will be ‘causing deaths’ | TechCrunch [Internet]. [cited 2020 Nov 29];Available from: <https://techcrunch.com/2019/06/12/why-vinod-khosla-thinks-radiologists-still-practicing-in-10-years-will-be-causing-deaths/>
118. Clinical radiology UK workforce census 2018 report [Internet]. 2019 [cited 2020 Nov 29]. Available from: https://www.rcr.ac.uk/system/files/publication/field_publication_files/clinical-radiology-uk-workforce-census-report-2018.pdf