Learning Structure-Function Interactions of the Heart using Generative Deep Learning Methods



Julius Ossenberg-Engels

St Cross College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2021

Abstract

Cardiovascular diseases account for the highest number of annual deaths worldwide, a burden exacerbated by current limitations in disease understanding. Accurate clinical outcome prediction is key for reducing these fatalities. However, high phenotype heterogeneity of diseases like hypertrophic cardiomyopathy and myocardial infarction (MI) complicates this task. Current clinical methods used for assessing outcome risk, such as ejection fraction, are often seen as too coarse and non-specific, failing to take complex variations of myocardial shape and function into account. Thus there is a growing need for more holistic approaches. The rise of deep learning has revolutionised medical image analysis and boosted interest in generative methods for cardiac modelling, due to the networks' inherent aptitude for capturing complex data distributions. The aim of this thesis was therefore to investigate their use in learning structure-function interactions in the heart to facilitate cardiac function modelling and aid with disease understanding and clinical outcome prediction.

This thesis shows that Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) can be used for synthetic 2D Cardiac Magnetic Resonance (CMR) data generation, reaching high levels of image realness that can convince trained observers. It further explores modes of variation within CMR data sets and uses latent space manipulation to guide the image generation process towards creating sub-population specific cardiac data. Conditional GANs (cGANs) are then used for 2D and 3D cardiac function modelling whereby high levels of predictive accuracy are achieved. Aided by the inclusion of patient metadata, subpopulation-specific cardiac contraction patterns are extracted for both healthy and diseased patient groups from which understanding of (patho)physiological differences between groups is derived. This thesis presents a state-of-the-art segmentation network for 2D long-axis UK Biobank CMR data using cGANs to achieve an epicardial Dice score of 0.971, thereby aiding in the development of 3D CMR reconstruction pipelines based on 2D slice segmentations. Additionally, a new state-of-the-art classification network for assessing the risk of developing Major Adverse Cardiac Events (MACE) in MI survivors is presented which achieves an accuracy of 74.8%. By using class activation maps, it thus enables the derivation of MACE-related pathophysiology understanding.

Overall the results presented in this thesis demonstrate the power of generative methods to capture structure-function interactions from CMR data and provide initial validation of their applicability in clinical problems.

Acknowledgements

First and foremost, I would like to thank my supervisor, Professor Vicente Grau, who has guided me wisely through the academic obstacles of the past 4 years. Vicente, I have always appreciated your calm, insightful and humorous nature during our meetings, especially at times when results were bad and frustrations high. Thank you for getting me to the finish line.

This thesis truly is a matter of the heart. With its submission a big chapter in my life comes to a close. I have spent nearly a decade in this beautiful city that I have been able to call home. A time marked by so many ups, and a few downs, though it will remain as a most cherished memory deeply lodged into my heart. Along the journey I had the opportunity to meet some of the most remarkable, inspiring and truly wonderful people. I would like to take this opportunity to thank a few of them here.

First of all, thank you to the staff on the ONBI program for making this PhD possible in the first place and guiding me through the first year. Particular thanks go to Peter & Lisa. To all the staff in the IBME, thank you for looking out for us (and leaving plenty of sandwiches out). Special thanks go to Denise for her unparalleled energy and to the Linux gods Ivan and Tom for keeping my computer (and thereby degree) patched up. Next up, the Cake Club. Thank you for being a great bunch of lab mates, terrible cake bakers (did anyone actually ever bake?), but much better debaters. I have learnt a lot through our numerous discussions and value you all highly. Just remember, Jorge, not every paper is straight up terrible. To my IBME lunch crew, thank you for letting me join your illustrious round, it made my lunch breaks immeasurably more entertaining and I would be hard pressed to think of a topic we did not discuss. Special shoutout to Navs for elbow patches, VP wisdom and being a good listener.

To my ONBI cohort, thank you for a brilliant first 6 months of countless caterpillar cakes, beard-offs and Christmas dinners. I wish you all the best with your future research! Also thanks to Nick for consistently turning up later than me. To the West Wing Legends, you truly are legends! Alex, Alex, Charity, Tilly, Ellie, Helen, Ben and Pieter-Jan, thank you for cold December surfs, Wine & Arguing Club meetings and constant chaos & disaster. Most importantly though, thank you for your continued friendship, you are all good eggs. To my St Cross friends, Allie,

Olivia, Jodie, Sorcha and all the other people I had the good fortune of befriending during my time as (V)P, thank you for one of the most enjoyable years in Oxford. Would not want to miss it! Thanks also go to my favourite porters Ben & Stephen and the StX staff. Lastly an honourable shout out to the Sexy Salmons! One day guys, one day. Big thanks to the MB20 lot, you are a great group of mates and I cannot wait for more London (and elsewhere) shenanigans now that I have a bit more time. To my Juxon Street compadres Eoin, Amy and Sagar, thank you for a year of Joe Wicks impressions, failing to teach me the value of early mornings and enough Oasis karaoke to last me a lifetime. It's been fun and I miss you.

To the Extended Crib, Nayana, Jess and Stuart, thank you for being a wonderful addition to my life and all the great moments we have had over the past years. May the charades forever continue. To my boys Flozza, Boo and Paddy, thank you for being the family away from family. It has been the utmost pleasure living with you on the island and the best way that lockdown could have been spent. I cannot wait for the day when we buy the house and move back in. Deal?!

To Anna, thank you for all your support and love on this last stage of the journey. Truly looking forward to the next chapter of my life and going on some real adventures with you, so lez go!

To Opa and Oma Dorle, and to the wider family, thank you for your support over the years. It means a lot to have you all in my corner.

And finally to Mama, Papa, Moritz and Max, thank you for all you have given me. I could not have done it without you. Your love, support and advice throughout my life is what got me here. So thank you. Yours truly, Lulu.

In loving memory of Nini, Heinz, Oma and Patrick Finlay.

Contents

List of Figures **x**

List of Tables **xviii**

1 Introduction **1**
1.1 Motivation . . . 2
1.2 Research Aims . . . 4
1.2.1 Overall Aim . . . 4
1.2.2 Specific Aims . . . 4
1.3 Thesis Structure & Key Contributions . . . 5

2 Background **7**
2.1 Cardiac Background . . . 9
2.1.1 Cardiac Anatomy and Structure . . . 9
2.1.2 Cardiac Function and the Cardiac Cycle . . . 11
2.1.3 Cardiac Function Parameters . . . 14
2.1.4 Cardiac Pathophysiology . . . 14
2.2 Cardiac Imaging . . . 18
2.2.1 Cardiac Imaging Modalities . . . 18
2.2.2 MRI . . . 21
2.2.3 CMR . . . 25
2.3 Deep Learning . . . 30
2.3.1 Introduction . . . 30
2.3.2 Neural Networks . . . 30
2.3.3 Convolutional Neural Networks . . . 35
2.4 Generative models . . . 47
2.4.1 Introduction . . . 47
2.4.2 GAN Introduction . . . 48
2.4.3 GAN Structure . . . 49
2.4.4 GAN Loss . . . 50
2.4.5 GAN Optimizer . . . 51
2.4.6 GAN Training . . . 52

2.4.7 GAN Problems . . . 53
2.4.8 Improvements to GAN Training . . . 56
2.4.9 GAN Performance Metrics . . . 56
2.4.10 GAN Developments . . . 58
2.4.11 GAN Outlook . . . 64
2.4.12 GANs in Medicine . . . 64
2.5 VAEs . . . 68
2.5.1 Introduction . . . 68
2.5.2 VAE Structure . . . 68
2.5.3 VAE Loss . . . 69
2.5.4 VAE Training . . . 71
2.5.5 VAE Reparameterization Trick . . . 71
2.5.6 VAE vs GAN . . . 72
2.5.7 VAE Variants . . . 73
2.6 CMR Analysis with Deep Learning . . . 75
2.6.1 CMR Segmentation . . . 75
2.6.2 CMR Quantification . . . 77
2.6.3 CMR Motion Analysis . . . 78
2.6.4 CMR Classification . . . 79
2.7 Literature Review Conclusion . . . 82

3 Healthy Cardiac Structure - How to Capture It 84
3.1 Introduction . . . 85
3.2 Data . . . 86
3.2.1 Introduction . . . 86
3.2.2 Pre-Processing Steps . . . 86
3.3 GANs for Capturing Cardiac Structure . . . 89
3.3.1 Introduction . . . 89
3.3.2 GAN Structure . . . 89
3.3.3 GAN Hyperparameter Optimisation . . . 90
3.3.4 GAN Autoencoder Pre-training . . . 104
3.3.5 "Optimum" GAN . . . 106
3.4 VAEs for Capturing Cardiac Structure . . . 112
3.4.1 Introduction . . . 112
3.4.2 VAE Structure . . . 112
3.4.3 VAE Hyperparameter Optimization . . . 113
3.4.4 "Optimum" VAE . . . 118
3.4.5 VAE Latent Space Analysis . . . 120
3.4.6 New Data Generation . . . 124
3.5 Conclusion . . . 127

4 Cardiac Motion - Healthy and Diseased **129**
4.1 CGANs for the Prediction of Cardiac Contraction 130
4.1.1 Disclaimer 130
4.1.2 Introduction 130
4.1.3 Related Works 132
4.1.4 Methods 133
4.1.5 Experiments 135
4.1.6 Results 136
4.1.7 Discussion 140
4.1.8 Conclusion 146
4.2 Metadata Inclusion 148
4.2.1 Introduction 148
4.2.2 Outline 148
4.2.3 Metadata vs No Metadata 149
4.2.4 Individual vs Joint Network 156
4.2.5 Multiple Metadata Inputs 158
4.3 Conclusion 163

5 From 2D to 3D: 2-Ch Long-Axis Segmentation **166**
5.1 Introduction 168
5.1.1 Disclaimer 168
5.1.2 3D networks in CMR 168
5.1.3 Moving from 2D to 3D 169
5.2 Data 171
5.2.1 Choice of Data 171
5.2.2 Segmentation Process 171
5.2.3 Additional Data 174
5.2.4 Pre-processing 175
5.2.5 Data Augmentation 175
5.3 Methods 178
5.3.1 Metrics 178
5.3.2 Network Architectures 178
5.4 Segmentation Network Comparison 179
5.4.1 Results 180
5.4.2 Discussion 180
5.5 Segmentation UNET Optimisation 181
5.5.1 Introduction 181
5.5.2 Hyperparameter Optimisation 182
5.5.3 Results 182

5.5.4 Discussion . . . 183
5.6 Addressing Myocardial Breakage . . . 188
5.6.1 Introduction . . . 188
5.6.2 Methods . . . 189
5.6.3 Results . . . 190
5.6.4 Discussion . . . 190
5.7 Conclusion . . . 193

6 3D Cardiac Function Modelling for MACE Risk Prediction 195
6.1 Introduction . . . 196
6.1.1 Motivation . . . 196
6.1.2 Overview of Methodology . . . 197
6.2 Data . . . 200
6.2.1 Introduction . . . 200
6.2.2 Pre-processing . . . 201
6.3 Part I: 3D Functional Modelling . . . 203
6.3.1 3D Network Comparison . . . 203
6.3.2 Hyperparameter Optimisation . . . 208
6.3.3 Part I Discussion . . . 211
6.4 Part II: MACE Classification . . . 215
6.4.1 Classification Methodology Overview . . . 215
6.4.2 Direct Linear Classification . . . 216
6.4.3 Classification CNN with ED & ES Ground Truth Data . . . 218
6.4.4 Classification CNN with ES GT & Prediction Data . . . 221
6.4.5 Part II Discussion . . . 226
6.5 Conclusion . . . 229

7 Conclusions & Future Work 232
7.1 Summary of Results & Contributions . . . 233
7.1.1 Capturing Cardiac Structure . . . 233
7.1.2 Modelling Cardiac Function in 2D and 3D . . . 234
7.1.3 Enabling a 2D to 3D Volume Reconstruction Pipeline . . . 235
7.1.4 Prediction of Clinical Outcomes . . . 236
7.2 Future Work . . . 237
7.2.1 Overcoming GAN & VAE Limitations for Improved Cardiac Structure Captioning . . . 237
7.2.2 Personalised Function Modelling . . . 238
7.2.3 Methodology Improvements . . . 239
7.2.4 Drive towards Clinical Implementation . . . 239

Appendices

A Supplementary Section for Chapter 4 **242**
A.1 Point Cloud Network Extension 242
A.1.1 Disclaimer 242
A.1.2 Introduction 243
A.1.3 Methods 243
A.1.4 Results 243
A.1.5 Conclusion 245

References **246**

List of Figures

2.1 From left to right: Cardiovascular system in the human body; Location and orientation of the heart in the thoracic cavity; Anatomical cross-section of the heart. Figures adapted from [6], [7] and [8]. . . . 10
2.2 Coronary vessels surrounding the heart and cross section of cardiac wall. Figure adapted from [10] and [11]. 11
2.3 Top: Pressure variation during the cardiac cycle in the aorta, ventricles and atria. Bottom: Schematic depiction of cardiac shape variation during the cardiac cycle. Figure adapted from [13]. 13
2.4 Changes in cardiac structure due to Dilated and Hypertrophic Cardiomyopathy.[16] 17
2.5 In clockwise direction, starting top left, examples of: Echocardiography, Cardiac CT and Cardiac PET. Figures adapted from [18], [19] and [20]. 19
2.6 Nuclear Magnetic Resonance principle. Left: Proton precession, Right: Net magnetization. Figure adapted from [21] and [22] 21
2.7 NMR Relaxation Process. Figure adapted from [23] 23
2.8 Left: Spin Echo Pulse Sequence. Right: Gradient Echo Pulse Sequence. Figure adapted from [24] 24
2.9 Cardiac Axes. Figure adapted from [26] 26
2.10 In clockwise direction starting top left: MR Tagging, 17-Segment Model and Myocardial Perfusion Imaging. Figure adapted from [30], [29] and [31]. 29
2.11 Simplified Neural Network Structure. Figure adapted from [35]. . . 31
2.12 Examples of common neural network activation functions. Figure adapted from [37]. 32
2.13 Top: Convolution operation. Bottom: Example CNN structure. Figure adapted from [40]. 34
2.14 a) Depiction of the Inception Block structure. Figure adapted from [43]. b) Depiction of ResNet structure. Figure adapted from [48]. . 39
2.15 a) Depiction of the Recurrent NN structure. Figure adapted from [54]. b) Depiction of DenseNet structure. Figure adapted from [53]. 40

2.16 Fully Convolutional Network (FCN) Architecture. Figure adapted from [65]. 43
2.17 UNET Architecture. Figure adapted from [66]. 43
2.18 Graphical depiction of maximum likelihood process. Figure adapted from [88]. 48
2.19 GAN structure. Figure adapted from [89] 49
2.20 Curve shows generator gradient for different values of discriminator performance on fake inputs for different cost functions and highlights their problems. Minimax game (original GAN cost function where $J(G) = -J(D)$) and Maximum Likelihood (which assumes an optimal discriminator) both suffer from vanishing gradients when D is initially good at identifying fake samples. The latter also receives almost all gradient updates from the right end meaning some batches will dominate training. The non-saturating heuristic method trains G to maximise $\log(D(G(z)))$ instead of minimizing $\log(1 - D(G(z)))$ which provides much larger gradients in earlier training, however, it can have unstable numerical gradients for G [103]. Figure adapted from [88]. 54
2.21 cGAN Structure. z is the latent generator input, c the conditional input and x a real sample. 59
2.22 GAN Architecture Variants. Figure adapted from [127]. 61
2.23 VAE Structure. Figure adapted from [171]. 69
2.24 Reparameterization Trick in VAEs; turning a stochastic node into differentiable node. Figure adapted from [176]. 71

3.1 Example images from the Kaggle Competition data set. Even though the data set has been pre-processed to improve image homogeneity, a diversity in image quality and content can still be observed. A range of example images that capture the diversity are displayed here. Cardiac mid-axial SAX slices can range from "normal" scans which capture the ventricles in full (a), including some of the surrounding anatomy, to scans where the heart occupies either a very small (b) or very large (c) fraction of the image space. Additionally scans that fail to display easily identifiable cardiac anatomies are also included in the data set (d-f). 88
3.2 Structures for Generator G and Discriminator D networks. 90
3.3 Example images from the qualitative rating assessment with scores of 1-5 in rows a)-e) respectively showing characteristic image features for each score. 94

3.4 Results of GAN hyperparameter experiments A-J. The red line indicates the FID score. The dark blue and dark green line represent the rating and classification scores respectively, as judged by the author, whilst the light blue and light green lines represent the rating and classification scores respectively, as judged by a group of experts. Note that all axes have been chosen so that better scores are at the top of each graph. The vertical dashed grey line indicates the base line value for each parameter prior to optimisation, whilst the vertical solid green line indicates the chosen optimum value. Optimum values are also summarised in the attached table. 99
3.5 Example images from hyperparameter experiments, including corresponding FID scores. Images have been chosen from different experiments to highlight the correlation between FID score and image quality & content. 100
3.6 Example image results from using an AE pre-trained generator in the GAN. 105
3.7 GAN training curves for “optimum” GAN training. 107
3.8 Example image results from “optimum” GAN. 108
3.9 a) Distributions of ratios of myocardium to cavity segmentation areas for real and fake images in “optimum” GAN. b) Distributions of cavity segmentation areas for real and fake images in “optimum” GAN. c) Distributions of myocardium segmentation areas for real and fake images in “optimum” GAN. All p-values are $<10^{-2}$ calculated using the KS-test, indicating a statistically significant difference between the distributions. 108
3.10 Results of VAE hyperparameter experiments A-F. The red line indicates the FID score. The dark blue line represents the validation loss. The vertical dashed grey line indicates the base line value for each parameter prior to optimisation, whilst the vertical solid green line indicates the chosen optimum value. Optimum values are also summarised in the attached table. 115
3.11 Example outputs from "optimum" VAE random Gaussian inputs. . 119
3.12 VAE outputs given one-hot encoded latent space inputs whereby only single latent space vector elements were activated at a time with equidistantly spaced values taken from a range of [-σ, σ], where σ is the standard deviation of the latent space distribution during test time. The outputs demonstrate different modes of change for identifiable anatomical features, namely LV Lumen Size, Myocardium Shape, Image Contrast and Apical to Basal Position, related to specific latent space vector elements. 122

3.13 Example VAE outputs given sequence of one-hot encoded latent space activations for vector element values in range [-σ, σ], where σ is the standard deviation of the latent space distribution during test time, when latent space dimension is of size 24. 123
3.14 Example VAE outputs given sequence of one-hot encoded latent space activations for vector element values in range [-σ, σ], where σ is the standard deviation of the latent space distribution during test time, when latent space dimension is of size 16. 123
3.15 Example VAE outputs given sequence of one-hot encoded latent space activations for vector element values in range [-σ, σ], where σ is the standard deviation of the latent space distribution during test time, when latent space has a high β weighting. 125
3.16 VAE output results of changing single element in latent space that is known to be associated with LV cavity size. The same random latent space inputs were augmented by changing the same element to 0 to produce VAE outputs A and then to +2 to produce VAE outputs B. 126

4.1 cGAN U-Net Generator architecture; Numbers indicate number of filters $\times$ image dimensions for each layer. The Discriminator architecture is equivalent to the encoding part of the Generator, followed by a softmax layer at the end. 134
4.2 Workflow diagram: ED Ground Truth (GT) frames are fed into cGAN generator to produce ES predictions from which MSE and SSIM are computed; Predictions and ES GTs are segmented by U-Net from which Dice and AEF are computed. 134
4.3 cGAN prediction of ES frame for Patient A, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0035, 0.89 & 0.91] respectively. 137
4.4 cGAN prediction of ES frame for Patient B, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0047, 0.85 & 0.94] respectively. 138
4.5 Subject Group Histograms; Y-axis= frequency; X-axis= Metric; Group A in green, Group B in blue 139
4.6 cGAN prediction of ES frame for HCM Patient A, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0075, 0.81 & 0.74] respectively. 140
4.7 cGAN prediction of ES frame for HCM Patient B, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0032, 0.86 & 0.67] respectively. 141

4.8 Segmentations of ES ground truth and ES prediction and difference between segmentations for high-BMI patient when applied to low-BMI trained network. 141
4.9 Average LV segmentation outputs of ES ground truth and ES predictions for Group B patients when applied to network trained on respective Group A data. The blue part indicates the areas where LV myocardium was found in ES predictions, but not in ES ground truths, whereas the yellow part indicates the opposite. 142
4.10 Standard cGAN generator architecture. 149
4.11 Standard cGAN discriminator architecture. 150
4.12 cGAN meta architectures: a) Standard cGAN architecture b) First Layer metadata inclusion c) Latent Layer metadata inclusion d) First+Latent Layer metadata inclusion e) All Layer metadata inclusion f) Discriminator First Layer metadata inclusion. 151
4.13 Comparision of standard (non-meta) cGAN with different meta cGAN architectures based on MSE, SSIM and DICE scores. The horizontal dashed black line indicates the score of the standard cGAN. Note that the y-axis on the MSE graph has been inverted. 154
4.14 Top left image: Average ES output for ED input cases given additional meta information of [0,0,0], where square bracket values indicate [gender, age, BMI].
All other images: Average difference between top left ES outputs with given [0,0,0] meta combination and ES outputs given the respective meta combination indicated. Dark areas indicate myocardium label predictions for [0,0,0] baseline case, yet cavity label predictions for the respective other case. White areas indicate the opposite. 160
4.15 Zoomed image of [0,1,0] case from Figure 4.14 indicating areas of over- and under-prediction of myocardium between the [0,0,0] and [0,1,0] meta ES predictions. The red arrow indicates an area that over-prediction between the [0,0,0] and [0,1,0] cases. Respectively the yellow arrow indicates an area of under-prediction. 160
4.16 Confusion matrix showing array of MSE differences between mean meta outputs in 4.14. All values are in 10^2. Higher values indicate larger difference between given combination of meta outputs. 161

5.1 a) Example of long-axis 2ch scan from the UK Biobank. b) Corresponding manually segmented endo- and epicardial contours, as well as basal contours. c) Full pixel labels for LV ventricle cavity (in gray) and myocardium (in white). d) Overlay of segmentation labels onto image. 172

5.2 Example of the systematic difference that can occur between pixel-wise and interpolation based segmentation methods. The interpolation method systematically labels myocardium pixels further outwards than the pixel-wise method. 173
5.3 Examples of papillary muscles as seen on 2ch long-axis scans. Papillary muscles are depicted as shaded regions in the LV cavity, indicated by the red arrows in a) and b). Due to their similarity in image intensity, they easily merge with the myocardium, particularly in frames close to ES, as seen in c) and d), and care needs as to be taken as to not include them in a myocardium segmentation. Using the whole image sequence to observe the motion of the myocardium hereby usually helps. 174
5.4 Examples for ED and ES long-axis 2ch segmentations from the UK Biobank using a pixel-wise segmentation method. LV cavity pixels are labelled in gray and myocardium pixels in white. 175
5.5 Examples of augmentation to expand training data set size, including rotation, translation and zoom. 177
5.6 Fully Convolutional Network developed by Bai et al. [204] for the segmentation of short-axis and long-axis CMR data from the UK Biobank. Figure taken from [204]. Successive layers of convolutions, batchnorm layers, ReLU activations and maxpooling layers are applied from which outputs serve both as the input to the next downstream layer, as well as being upsampled and concatenated to form the final layer. 179
5.7 Examples of segmentations which contain a breakage in the myocardium segmentation, as pointed to by the yellow arrows. For most cases this occurs around the apex where the myocardium is thinnest, as seen in a), however, breakages can also occur further up the z-axis. 181
5.8 Results of UNET hyperparameter optimisation experiments A-E. The red line indicates the fraction of test cases that showed a breakage in the myocardium segmentation (Note that the axis is flipped upside down to display low breakage values at the top of the graph). The light blue line indicates the Dice score of the combined cavity and myocardium labels, whilst the dark blue and green lines indicate the Dice scores of cavity and myocardium labels respectively. The gray dashed vertical line indicates the baseline parameter, whereas the green solid line indicates the best performing parameter value. The attached table gives a summary of all optimum values. 184

5.9 Example of a segmentation which contains a breakage in the myocardium segmentation, as pointed to by the green arrows. For most cases this occurs around the apex where the myocardium is thinnest. a) Original CMR image b) segmentation with breakage at apex c) breakage resulting in two distinct pixel areas and labels for myocardium (red and yellow). 188
5.10 Results of cGAN for 2ch LAX segmentation using different adversarial loss weightings. a) Displays the Dice scores for combined myocardium and cavity (blue), cavity (light blue) and myocardium (navy blue), as well as the myocardium breakage fraction (red) for 11 different weighting factors. b) is a zoomed version of a) focusing on just the first 5 weighting factors. Note that the y-axis for Breakage is flipped upside down to display low breakage values at the top of the graph. The gray dashed vertical line indicates the baseline parameter, whereas the green solid line indicates the best performing parameter value. 190

6.1 Overview of Methodology. Part I compares suitability of 3D UNET, 3D cGAN and 3D VAE architectures in predicting 3D ES volumes from ED volume inputs based on metrics including Dice Scores, Hausdorff Distance and EFs. Part II deals with MACE classification through a) direct classification given Part I metrics, b) a classification CNN given ES & ED ground truth data with additional EF input or loss and c) a classification CNN given the differences between ES predictions and ES ground truths. This is additionally used for Class Activation Mapping (CAM) to investigated MACE-related pathophysiology. 198
6.2 MACE data distribution and data pre-processing steps. These include transforming 3D myocardial and cavity meshes into 3D segmentation voxel grids. Volume augmentations in the form of random rotations and translations were then applied to increase the data set. 201
6.3 3D UNET structure. 204
6.4 3D cGAN Discriminator structure. 204
6.5 3D UNET Hyperparameter Optimisation Results. Validation loss results for different numbers of layers, numbers of convolutions, filter scaling factors, dropout levels and learning rates are shown. The vertical grey, dashed lines indicate the baseline value for each parameter whilst the vertical green, solid lines show the optimum values. 210

6.6 3D mesh representations of 3D UNET prediction results. The averages of ES myocardium predictions (in green) and corresponding ES ground truths (in white) are shown for both non-MACE and MACE test sets along the four long-axis viewpoint directions. The arrows indicate regions of over-and underprediction for the non-MACE and MACE case. Using g) as an example, the endocardial cavity contour of the ES prediction is indicated by the yellow arrow. The endocardial contour of the ES ground truth sits below that, i.e. the line bounding the inner white mesh. 212
6.7 Receiver Operator Characteristic (ROC) curves for direct classification of non-MACE vs MACE subjects using different metrics derived from Stage I function modelling. The Area under the Curve (AUC) for each metric is given in the figure legend. Metrics include Combined Myocardium and Cavity Dice Score, Cavity Dice Score, Myocardium Dice Score, Hausdorff (HD) Distance, difference between predicted and ground truth EF values and ground truth EF values. 217
6.8 MACE Classification Network structure. 219
6.9 MACE Classification Network Variants. a) Standard structure with no additional EF information. b) Structure with additional EF regression loss. c) Structure with Additional EF input 220
6.10 2D short-axis and long-axis slices of the average test set ES myocardium segmentation volumes at different positions along the z- and x-axis respectively are shown together with associated class activation maps (CAMs). These highlight the activated regions in the last convolutional layer of the MACE classification network and show which parts of the input data the network is focusing on when making classification decisions. For orientation the respective position of the RV is indicated on the short-axis slices. For the long-axis slices the RV is positioned out of the page relative to the 2D slice. 224
6.11 Activation maps from 6.10 rendered onto a 3D ES myocardium volume and shown from all four viewpoints on the x-y grid, both from the side and the top. The colours as given by the colour bar indicate the levels of activation. The location of the RV is also labelled. 224

A.1 ES prediction results for two sample cases of the unseen test data set. Figure is courtesy of Marcel Beetz. 244

List of Tables

3.1 GAN hyperparameters with associated value ranges and experiment number corresponding to results in Figure 3.4. Baseline values for each parameter are indicated in bold. 91

3.2 VAE hyperparameters with associated value ranges. Baseline values for each parameter are indicated in bold. 114

3.3 Comparison of quantitative and qualitative results for optimum GAN and VAE performance in capturing cardiac structure. Note that lower Minimum FID scores, higher Qualitative Rating Scores and lower "Fake" Classification scores indicate better network performance. 120

4.1 MSE (in 10^{-3}s), SSIM, Dice scores and AEF differences for patient groups. Note that lower MSE and AEF, and higher SSIM and Dice, indicate better predictions. 138

4.2 MSE scores (in 10^{-3}s) for different network architectures when given different meta types as meta input. Note that lower MSE indicate better predictions. Best overall average score and best scores for each architecture across the three groups are given in bold. 153

4.3 SSIM scores (in 10^{-1}s) for different network architectures when given different meta types as meta input. Best overall average score and best scores for each architecture across the three groups are given in bold. 153

4.4 Cavity DICE scores for different network architectures when given different meta types as meta input. Best overall average score and best scores for each architecture across the three groups are given in bold. 153

4.5 Myocardium DICE scores for different network architectures when given different meta types as meta input. Best overall average score and best scores for each architecture across the three groups are given in bold. 154

4.6 MSE, SSIM and DICE scores for two networks. A) Training a non-meta standard cGAN on just female subjects, then testing separately on female and male subjects. B) Training a First Layer meta cGAN on a mixed population of male and female subjects, then testing on a mixed population of male and female subjects separately with correct and incorrect gender meta labels. 157

5.1 Comparison of architectures for 2ch LAX segmentation. Networks based on architectures by Bai et al. [204], Xu et al. [269] and a proprietary UNET were tested and compared. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for all, ED&ES, ED and ES test cases. . . 181
5.2 UNET Hyperparameter list and associated value ranges. Baseline values are given in bold. 182
5.3 Testing the previously trained UNET on data from Dr Lukaschuk. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for ED&ES, ED and ES cases. 183
5.4 Re-training UNET with additional data form Dr Lukaschuk and testing on original test set. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for all, ED&ES, ED and ES cases. 183
5.5 Segmentation results of the cGAN when using baseline parameters and when using a PatchGAN setup. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for all, ED&ES, ED and ES cases. 191

6.1 Stage I performance results from 3DUNET, 3DcGAN and 3DVAE in prediction ES segmentation volumes based on ED segmentation inputs. Results are given for non-MACE (nM), MACE (M) and combined non-MACE and MACE (Comb) test cases for Combined Myocardium and Cavity Dice, Cavity Dice, Myocardium Dice, Hausdorff Distance and Absolute Difference in EF scores. The values in bold indicate the best value for each test group and metric among the three networks. 205

6.2 P-values for difference in performance between non-MACE and MACE test sets in table 6.1. Calculated using the Kolmogorov-Smirnov test. 206
6.3 3D UNET hyperparameters and associated value ranges. Baseline values are given in bold. 208
6.4 Optimised 3D UNET metric results and associated p-values. 211
6.5 AUC Scores of MACE Classification Network Variants. 220

1 Introduction

Contents

1.1 Motivation **2**
1.2 Research Aims **4**
1.2.1 Overall Aim 4
1.2.2 Specific Aims 4
1.3 Thesis Structure & Key Contributions **5**

1.1 Motivation

In recent decades, cardiovascular diseases (CVDs) have established themselves as the leading cause of death, accounting for 31% of global annual fatalities in 2017 [1]. Ranking this highly naturally comes with a hefty price tag for countries around the world, both economically and socially, and places a giant burden on healthcare systems. This renders CVDs a problem worth tackling through research.

The reason for CVDs claiming such a large proportion of total deaths is partially due to the disappearance of other, previously more common causes of death such as infections like polio. Equally contributing are recent changes in people's lifestyles which have led to an increased abundance of CVD risk factors such as diabetes and obesity. However, fundamentally the reason for the top spot lies in the continued lack of understanding of many such diseases.

In a clinical setting this knowledge gap is an enormous problem as it too often leads to inaccurate and unspecified diagnoses of CVDs. As a result of this, imprecise disease prognosis and uncertain outcome risk estimations occur. This means that treatments, even though medically available, are often either applied incorrectly or not applied at all out of fear of potential mistreatment. Accurate risk assessment of clinical outcomes is particularly important in cases where preemptive treatments are necessary such as giving implantable defibrillators to patients who might be prone to suffering from sudden cardiac death. Failing to achieve that results in sub-optimal patient care and more deaths. Hence a more profound understanding of CVDs would allow for more accurate and earlier disease diagnosis and in turn for more effective patient treatment and fewer CVD related deaths.

The reason why this is difficult to achieve is because many CVDs have a wide range of phenotypes and the relationship between such phenotypes and clinical outcomes is poorly understood. Especially diseases with a high phenotype heterogeneity such as hypertrophic cardiomyopathy (HCM) are a particular challenge. HCM can present with a wide range of structural cardiac abnormalities such as myocardial wall thickening or aortic outflow obstructions. Structural abnormalities like such can affect cardiac function in different ways and are often patient-specific. Based

on genetic and epigenetic factors, or caused through trauma or other remodelling processes, these changes affect the contraction of the myocardial tissue and how the heart functions as a whole.

Similarly, myocardial infarctions (MI) are an issue. Whilst being a very common diagnosis, the accuracy of long-term prognoses for MI patients is still very poor. Clinical metrics like ejection fraction are commonly used in practice, however, they are often seen as too crude for accurately assessing cardiac function in MI survivors.

Since cardiac function is largely what determines clinical outcomes in CVDs, a better understanding of how structural changes in the heart affect cardiac function is needed. Understanding those relationships will enable more specific prediction of cardiac function and in turn more accurate outcome estimation and patient treatment.

Whilst structure-function relationships have been studied, this has mostly occurred in isolated approaches typically examining specific global parameters and clinical biomarkers (e.g. cardiac wall thickness, ventricular volumes or ejection fraction (EF)). Although such biomarkers are clinically established, it is generally agreed in the field that these rather simplistic derivations from isolated measurements are often too coarse and general for accurate clinical prognosis [2, 3]. To better understand these relationships and to subsequently improve patient outcome predictions, a more comprehensive and holistic approach is necessary.

This is where neural networks come into play, in particular the subset of generative methods. Generative networks are highly suited for capturing complex underlying data distributions. In particular with the introduction of the Generative Adversarial Network (GAN) [4] in 2014, the applications of generative networks in the medical field have exploded. Conditional extensions of many such networks are of particular interest as they allow for the modelling of cardiac function on the condition of structural inputs, thereby enabling a study of the relationship between them.

Cardiovascular diseases present a serious medical challenge to society and need to be addressed. In order to do so, a better understanding of them is required, in particular of the relationship between structural changes and cardiac function. In

the dawn of deep learning, generative methods are potentially providing powerful tools for addressing those challenges and their use in this task requires investigation.

1.2 Research Aims

1.2.1 Overall Aim

The overall aim of this research is therefore to investigate the use of deep generative networks for gaining a more comprehensive understanding of structure-function interactions in the heart, given a variety of CMR data sets from healthy and diseased patients, and to extract sub-population specific understanding of cardiac function and pathophysiology in order to generate a basis for disease phenotype and clinical outcome classification.

This work should be seen as both a validation of generative methods in the field of cardiac image analysis and specifically cardiac function modelling, as well as a contribution towards the development of deep learning based clinical tools for disease prognosis and patient care.

1.2.2 Specific Aims

- To investigate the suitability of deep generative methods for learning healthy and diseased cardiac structure from 2D CMR image sets and to use such understanding to develop a pipeline for generating patient sub-population and data type specific synthetic CMR data for the use in other deep learning training applications.

- To investigate the suitability of conditional generative methods for learning cardiac function given structural 2D CMR input data for healthy and diseased patient groups.

- To further investigate how sub-population specific cardiac function can be modelled, through the use of additional patient metadata or otherwise, and how healthy and diseased sub-population specific cardiac physiology can thereby be extracted.

- To investigate the use deep generative methods for developing a 2-chamber long-axis CMR data segmentation network for the use on UK Biobank 2D CMR data in order to facilitate a wider 2D to 3D data transformation pipeline.
- To investigate the suitability of generative methods for learning 3D cardiac function given structural 3D segmentation data input for healthy and diseased patient groups.
- To investigate the suitability of deep classification networks and parameters derived from 3D functional modelling for the prediction of Major Adverse Cardiac Event risk after myocardial infarction and the understanding of related pathophysiology.

1.3 Thesis Structure & Key Contributions

This thesis presents a series of applications of deep generative methods in the field of cardiac image analysis with novel contributions made to address the research goals outlined in section 1.2.2. Below is an overview of the thesis structure and the key contributions from each chapter.

- **Chapter 1** explains the motivation for this research, outlines its aims and provides a chapter by chapter overview of key contributions to address them.
- **Chapter 2** summarises the relevant medical and technical background, covering cardiac anatomy and physiology, deep learning and generative networks with a particular focus on Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). Furthermore it provides an overview of the relevant literature.
- **Chapter 3** addresses the problem of capturing cardiac structure using generative networks. Hereby the suitability of GAN and VAE architectures for such tasks is explored and their performance is optimised. Furthermore, network latent space analysis is used for understanding modes of change

within CMR data and thereby enable patient specific generation of synthetic 2D CMR data.

- **Chapter 4** investigates cardiac function modelling in 2D by applying conditional GANs (cGANs) for end-diastolic (ED) to end-systolic (ES) CMR frame-to-frame prediction. Function that is specific to healthy and diseased patient sub-populations is hereby explored and physiological differences between patient groups are extracted. The inclusion of patient metadata such as age, sex and BMI into the modelling process is also investigated in its use for improving patient sub-population specific cardiac function prediction.

- **Chapter 5** proposes a cGAN based segmentation method for 2ch long-axis UK Biobank CMR data to facilitate a future 2D to 3D volume reconstruction pipeline that outperforms current methods. Different methods are investigated, in particular with regards to optimising segmentation accuracy and reducing problems of myocardium segmentation label breakages.

- **Chapter 6** investigates the use of 3D generative methods for 3D cardiac function modelling by using ED to ES volume-to-volume predictions, akin to the 2D methods explored in chapter 4. An optimised method is then used to explore functional differences between myocardial infarction (MI) survivors who have subsequently experienced Major Adverse Cardiac Events (MACE) or not. Lastly a state-of-the-art deep learning based classification method is developed for assessing MACE risk through which understanding of MACE related pathophysiology is derived.

- **Chapter 7** gives a final summary of the research and outlines its key contributions and limitations. It then discusses potential directions of future research.

2
Background

Contents

2.1 Cardiac Background 9
2.1.1 Cardiac Anatomy and Structure 9
2.1.2 Cardiac Function and the Cardiac Cycle 11
2.1.3 Cardiac Function Parameters 14
2.1.4 Cardiac Pathophysiology 14
2.2 Cardiac Imaging 18
2.2.1 Cardiac Imaging Modalities 18
2.2.2 MRI 21
2.2.3 CMR 25
2.3 Deep Learning 30
2.3.1 Introduction 30
2.3.2 Neural Networks 30
2.3.3 Convolutional Neural Networks 35
2.4 Generative models 47
2.4.1 Introduction 47
2.4.2 GAN Introduction 48
2.4.3 GAN Structure 49
2.4.4 GAN Loss 50
2.4.5 GAN Optimizer 51
2.4.6 GAN Training 52
2.4.7 GAN Problems 53
2.4.8 Improvements to GAN Training 56
2.4.9 GAN Performance Metrics 56
2.4.10 GAN Developments 58
2.4.11 GAN Outlook 64
2.4.12 GANs in Medicine 64
2.5 VAEs 68
2.5.1 Introduction 68

2.5.2 VAE Structure 68
2.5.3 VAE Loss 69
2.5.4 VAE Training 71
2.5.5 VAE Reparameterization Trick 71
2.5.6 VAE vs GAN 72
2.5.7 VAE Variants 73
2.6 CMR Analysis with Deep Learning 75
2.6.1 CMR Segmentation 75
2.6.2 CMR Quantification 77
2.6.3 CMR Motion Analysis 78
2.6.4 CMR Classification 79
2.7 Literature Review Conclusion 82

2.1 Cardiac Background

2.1.1 Cardiac Anatomy and Structure

The heart is an organ in the human body that serves as a biomechanical pump in the body's double circulatory system, which includes the pulmonary and systemic cardiovascular systems. Its primary function is to ensure the controlled flow of blood around the body and thereby enable the transport of oxygen and nutrients to all bodily organs. At the same time it facilitates the removal of metabolic waste products like carbon dioxide.

As seen in Figure 2.1, the heart is situated medially in the thoracic cavity between the lungs and the sternum and is oriented with its primary long axis (LAX) at a slight angle to the sagittal and coronal planes of the body. Its long axis runs from the top of the heart, also known as the base, to the bottom of the heart, also known as the apex and thereby roughly aligns with a directional vector running from the right shoulder to the left hip.

The whole heart is enclosed by the pericardium, a fibrous sac that protects the heart and prevents excessive enlargement. The pericardial space contains interstitial fluid which acts as a lubricant for cardiac motion. The average size of a healthy adult human heart is 12cm in length, with 8cm and 5cm in the short-axis dimensions. It has an average weight of 250-350g [5]. The actual size and weight can vary greatly based the gender, height, body weight and lifestyle of a person.

Structurally, the heart is partitioned into four chambers: the left atrium (LA), the right atrium (RA), the left ventricle (LV) and the right ventricle (RV). The RA is connected to the inferior and superior vena cava which return deoxygenated blood from the brain and other organs of the body to the heart. It is furthermore connected to the RV via the tricuspid valve. The RV in turn is connected to the pulmonary system via the pulmonary valve and pulmonary artery. Deoxygenated blood passes from the RV to the lungs where it gets oxygenated. It then enters the LA through the pulmonary vein. The LA is further connected to the LV through the mitral valve. From there the oxygenated blood is sent through the aortic valve

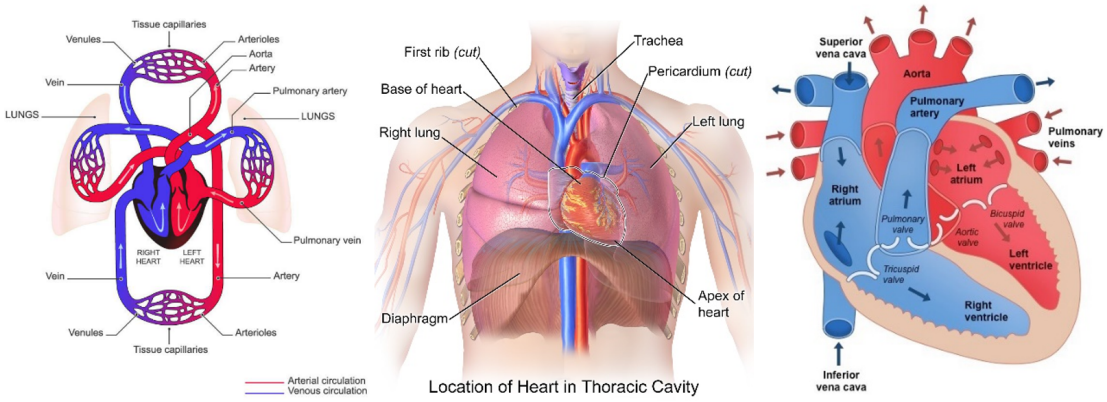


Figure 2.1: From left to right: Cardiovascular system in the human body; Location and orientation of the heart in the thoracic cavity; Anatomical cross-section of the heart. Figures adapted from [6], [7] and [8].

and the following aorta into the body. The left and right side of the heart are separated by the septum which keeps oxygenated and deoxygenated blood from mixing. Atria and ventricles work in temporal sequence while the connective valves ensure a unidirectional blood flow along a pressure gradient. The valves are held in place by papillary muscles which are attached to the heart walls in the ventricles. These contract with the rest of the ventricle and thereby prevent the valves from being pushed into the atria during ventricular contraction.

In the LV, two papillary muscles, the posterior and anterior muscles, are present. In addition, ridges of muscles called trabeculae also line the inside of the ventricular walls and aid with contractility and conduction. The LV approximates a conical shape along its long axis, though its short-axis cross-section follows more of an oval than a circular shape. During contraction this shape changes more drastically at the base and apex.

As seen in Figure 2.2, the heart wall is composed of three layers: the endocardium, the myocardium and the epicardium [9]. The endocardium is the innermost layer and mainly consists of endothelial cells and loose connective tissue, similar to the inner lining of blood vessels. It lines the cavities and valves of the hearts and forms a smooth and non-adherent surface between blood and tissue. The myocardium is by far the thickest layer and functions as the contractile component. It is primarily made up of cardiomyocytes or muscle cells and is thickest in the LV. The epicardium

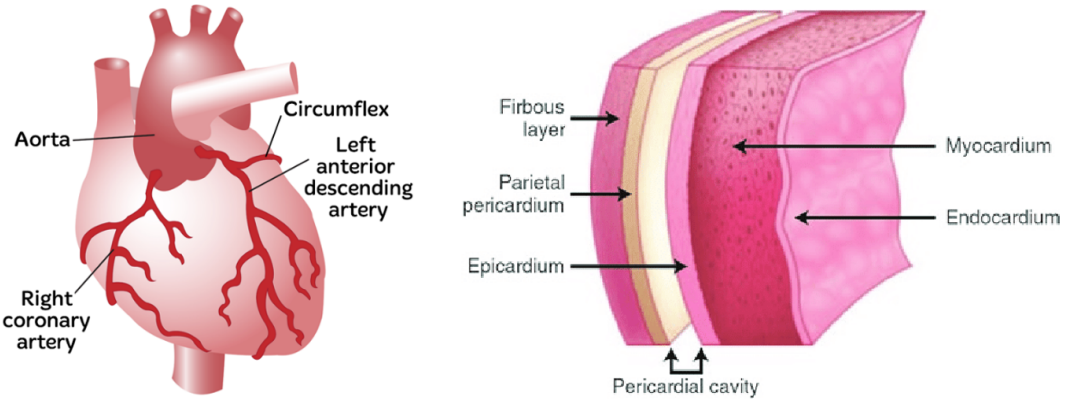


Figure 2.2: Coronary vessels surrounding the heart and cross section of cardiac wall. Figure adapted from [10] and [11].

forms the outer layer of the wall. It is made up of connective tissue and fat and acts as the outer protective layer of the heart.

Surrounding the heart are the coronary vessels which supply the myocardium with blood. The left and right coronary arteries branch off behind the aortic valve and supply the left and right ventricles respectively. Deoxygenated blood is returned through coronary veins. Occlusions in the coronary arteries on one side of the heart cannot be compensated by those of the other side, hence such impairments are extremely dangerous and need to be addressed immediately.

2.1.2 Cardiac Function and the Cardiac Cycle

The term "cardiac cycle" describes the processes that take place during a single sequence of contraction and relaxation of the heart. The contraction phase, usually referred to as systole, lasts for about a third of the time of the total cycle when the body is at rest. The relaxation phase, or diastole, which involves the relaxation of the myocardium and the filling of the chambers occupies the rest of the time. During exercise, the period for diastole shortens, whilst the time for systole usually remains relatively constant. Figure 2.3 shows the cardiac cycle in more detail.

During diastole all chambers are relaxed, the atrioventricular valves are open and the pulmonary and aortic valve, also known as semilunar valves, are closed. Blood passes through the atria into the ventricles, filling them up. The cycle starts

when an electrical pulse is created by the sinoatrial node located near the base of the heart in the RA. These cells act as the heart's natural pacemaker. Upon firing, the electrical signal travels through the atria and reaches the atrioventricular node where it causes a cascade of signals in the atrial wall, resulting in atrial contraction. This forces blood into the ventricles. When the body is at rest, ventricular filling is mostly passive during diastole, however, during exercise, this process gets helped along by atrial contraction. After a short delay which allows the atria to fully contract, the electrical impulse travels further down the heart through a collection of cells called the bundle of His located in the septal wall. From there it travels through the Purkinje fibres which branch out and around the apex of the ventricles.

Systole starts when the ventricular myocardium begins to contract in response to the electrical stimulus, causing the pressure in the ventricles to rise from its end-diastolic pressure of <10mmHg. By this point the ventricular volume will have reached its end-diastolic volume (EDV) somewhere between 120-140ml, depending on heart size. The value of the EDV is an important indicator of subsequent contraction strength, as given by Starlings law [12]. As the ventricular pressure rises, the atrioventricular valves shut, preventing any backflow of blood into the atria. During ventricular contraction, the ventricles depolarise. While the ventricular pressure is below the arterial pressure, isovolumetric contraction occurs which can lead to the atrioventricular valves bulging inwards into the atria. Once arterial pressures have been reached, the aortic valve opens. As blood rushes out, ventricular pressures increase further before dropping again, as seen in Figure 2.3.

When the ventricular pressure drops below the aortic pressure, the semilunar valves shut again, creating the second typical heartbeat sound. As the ventricular myocardium relaxes, the ventricular pressure drops further. Once it falls below the atrial pressure, the atrioventricular valves re-open and ventricular filling is restarted, signalling the beginning of diastole. This process lasts until the next electrical pulse is sent from the sinoatrial node which indicates the beginning of another systole and the end of a full cycle.

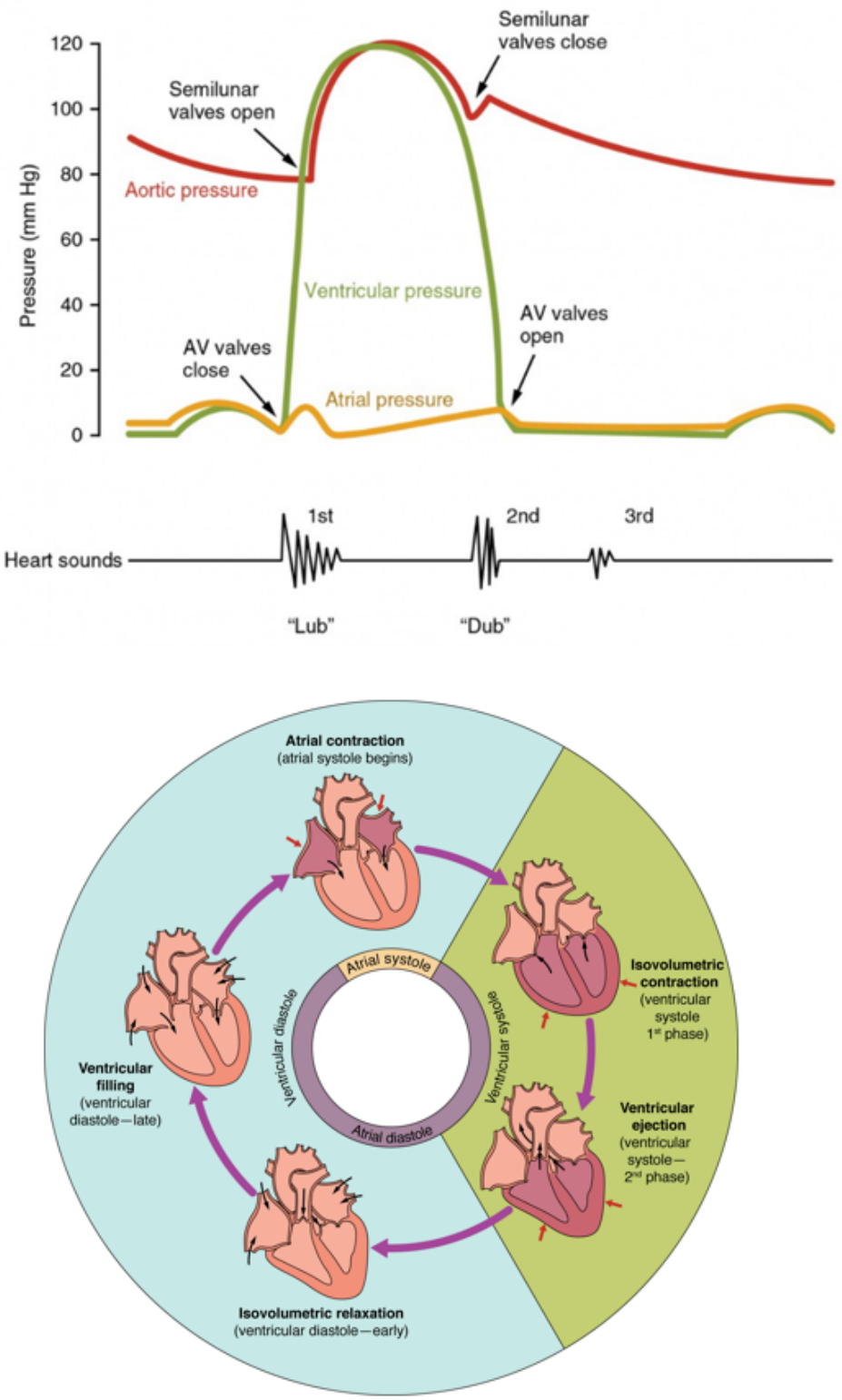


Figure 2.3: Top: Pressure variation during the cardiac cycle in the aorta, ventricles and atria. Bottom: Schematic depiction of cardiac shape variation during the cardiac cycle. Figure adapted from [13].

2.1.3 Cardiac Function Parameters

Cardiac function is parameterized by several clinical indices which are primarily derived from the end-diastolic volume (EDV) and the end-systolic volume (ESV). As the name suggests, these are the volumes of the LV at the two extreme points in the cardiac cycle, diastole and systole. They are either measured directly from 3D imaging data, or indirectly estimated based on 2D images with the use of Simpson's rule. This states that for each 2D slice of the heart along the long axis, the width and depth of the cavity along the short axis form the diameters of an ellipse at that location. Given the cross-sectional area of the ellipse of each slice and its thickness along the long axis, a total cavity volume can be computed when integrating along the long axis. This ventricular volume is, however, just an approximation, as the ventricle is neither of regular ellipsoidal shape, nor does the method account for partial volume effects.

EDV and ESV can be used to calculate stroke volume (SV) using the equation SV = EDV - ESV. From this, an estimation of cardiac output (CO) can be generated according to CO = SV x HR, where HR is the heart rate. Cardiac output quantifies the volume of blood in ml that is being ejected from the LV per minute. Another commonly used clinical metric is ejection fraction (EF). This represents the fraction of the EDV that is pumped out from the LV during each systole and calculated as EF = EDV-ESV/EDV = SV/EDV. Healthy EF values range between 0.5-0.7. Values below or above this range may indicate cardiac pathologies. Other measurements that can be taken are myocardial mass, myocardial strain or myocardial thickness.

During the cardiac cycle the atria and ventricles contract, causing the heart to change its overall shape. The motion occurs primarily along its short axis in a form of cross sectional shrinking or radial wall movement. The heart also contracts longitudinally and rotates around its long axis.

2.1.4 Cardiac Pathophysiology

Cardiovascular pathologies account for the highest number of annual fatalities worldwide with about 31% of all deaths. They encompass a wide range of diseases

which can be sub-classified into electrical, circulatory and structural defects. Usually these are caused by genetic predispositions or behavioural and lifestyle choices.

Arrhythmias

Cardiac arrhythmias are a common form of malfunction in the electrical system of the heart. The sinoatrial node usually acts as the heart's pacemaker and ensures a controlled and adaptive heartbeat. In arrhythmias, this rhythm gets compromised, causing the heart to beat either too quickly, too slowly, or with an irregular rhythm. Causes for arrhythmias are strongly related to structural changes in the myocardium, for example by the presence of scar tissue or wall thickening. Arrhythmias can lead to anything from mild pain to death, in the case of sudden cardiac arrest or ventricular fibrillation. Furthermore, arrhythmias are associated with an increased risk of thrombosis, or blood clotting, which in turn increases the risk of strokes.

Coronary Artery Disease

Coronary artery disease (CAD), or ischemic heart disease, is a form of circulatory disease which occurs when there is insufficient blood flow through the coronary arteries. This results in a lack of oxygen supplied to the myocardium, a state known as hypoxia, which can cause severe damage to the cardiac tissue. The reduced arterial blood flow is usually a result of atherosclerosis, a process that describes the deposition of fat, cholesterol and other substances, together known as plaques, in the arteries. Plaque build-up results in the gradual narrowing of the vessels. In its milder forms, CAD can result in angina which refers to chest pain caused by the short-term lack of oxygen to parts of the myocardium. If hypoxia persists for longer periods of time, myocardial cell death can occur in which case it is known as a myocardial infarction (MI) or heart attack. As a result, the myocardium loses its ability to contract and pump blood effectively. If this happens over a large region in the heart, it can result in death. Less severe cases will cause tissue scarring which can have long-term effects on cardiac health. Another risk with atherosclerosis is related to plaques directly which can cause harm if they become dislodged and travel to smaller vessels which they then fully occlude. Depending on the vessel

this can cause further infarctions or strokes. Alternatively this occlusion can be caused by blood clots which are also more likely to occur in plaque regions.

Structural Cardiac Pathologies

Structural cardiac pathologies are related to structural defects that are either congenital or have developed over time through wear and tear. These include heart valve diseases which can cause the valves to either become rigid and unable to fully open (stenosis) or to become loose and unable to properly close and seal, allowing blood to leak or regurgitate. Congenital heart diseases such as atrial septal defects can also occur, whereby the wall separating the atria contains holes that allow oxygenated and non-oxygenated blood to mix.

Hypertrophic Cardiomyopathy

Cardiomyopathy is a structural disease of the myocardium which affects its ability to effectively contract and pump blood. It presents with several forms among which hypertrophic cardiomyopathy (HCM) and dilated cardiomyopathy (DCM) are the most common. HCM is a cardiovascular pathology that occurs in 1 in 500 people worldwide [14]. It is diagnosed in patients as a hypertrophied non-dilated LV without there being evidence of any other cardiac or systemic disease that could have otherwise caused this [15]. It is a monogenetic disease which can be caused by a single mutation in 11 or more genes encoding the contractile myofilament protein components of the sarcomere [15]. As an autosomal dominant disease, offspring of affected patients have a 50% chance of developing the disease. It is therefore common practice for family members of diagnosed HCM patients to undergo genetic testing. Although genetic testing can be used to confirm positive genotypes, the relation between genetic mutation and clinical outcome has not yet been established, or at least it has proven unreliable due to strong phenotype heterogeneity [15]. Furthermore, if an exam-warranting family history does not exist, HCM often goes undiagnosed until the later stages of the disease. This lack of clinical recognition and the large number of undiagnosed patients is sharply contrasted by the prevalence of the disease, making research into this field highly relevant.

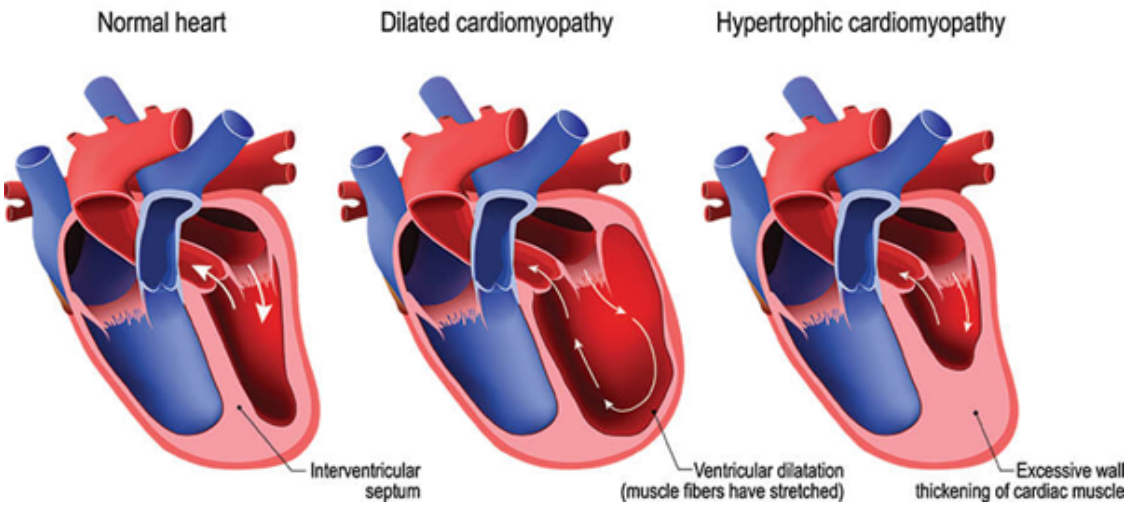


Figure 2.4: Changes in cardiac structure due to Dilated and Hypertrophic Cardiomyopathy.[16]

Diagnosis rarely occurs as part of a routine examination, but generally follows the onset of symptoms or other indicators such as an exam-warranting family history. The disease is usually diagnosed through CMR or echocardiography during which an enlargement of the ventricular wall of >15mm is observed [14]. Other forms of diagnostic testing such as ECG, physical exam and lab studies (excluding genetic testing) are neither sufficient nor necessary for HCM diagnosis. Cardiac magnetic resonance (CMR) imaging is the gold standard for diagnosis as it is much more accurate at diagnosing and characterizing HCM than echocardiography [15]. HCM patients present with high phenotype heterogeneity and can be grouped into three main symptomatic groups. Firstly, patients who are genotype positive but phenotype negative [15]. These are usually younger patients and present a higher risk of Sudden Cardiac Death (SCD). Secondly, patients that present without LV hypertrophy, but instead show other abnormalities such as myocardial fibrosis. Lastly, patients that display LV hypertrophy and potentially also LV outflow obstructions and other abnormalities.

HCM can occur at any age. The annual mortality rate sits at about 1% [17]. Disease pathways are varied and include sudden cardiac death (SCD), progressive heart failure (HF) with preserved systolic function with or without outflow obstruction and occasional systolic dysfunction and atrial fibrillation (AF)

with risk of progressive HF symptoms and embolic stroke [17]. SCD is more prevalent in younger patient populations (<30 yrs). It occurs without warning signs and generally after vigorous physical exertion. Though not always the case, HF tends to progress in most HCM patients and undergoes several stages with increasingly stronger ventricular outflow obstructions, higher outflow gradients and lower ejection fractions [14].

One of the major problems in the HCM space is risk stratification. The prognosis of a patient determines the choice of treatment. However, accurate prognosis of clinical outcomes has proven difficult. This is a particularly big issue with regards to SCD. If left untreated, SCD is fatal and occurs without warning. Effective treatments usually involve implantable defibrillators. However, the implantation surgery as well as the device itself hold some risk [14]. Therefore, performing this step on an otherwise asymptomatic heart must be well considered. Current risk assessment strategies involve risk-factor algorithms [14]. Risk factors such as family history or specific physiological measurements are taken into account. However, these algorithms are incomplete as they do not assign specific weightings to each factor and are therefore not personalized and completely accurate. About 0.5% of all HCM patients deemed to be at low risk die of SCD each year [15]. With the right treatment a lot of patients can reach a normal age. Treatment options for SCD and HF other than the aforementioned defibrillator include beta blockers and other arrhythmia drugs, surgical ablation to remove outflow obstructions, LV assistant devices (LVAD) and finally heart transplants [14].

2.2 Cardiac Imaging

2.2.1 Cardiac Imaging Modalities

A range of cardiac imaging methodologies has been developed over the past decades which enable clinicians and researchers to better understand and diagnose cardiac pathologies. In particular with regards to structural and functional analyses, there are four main methods that are common in clinical practise: echocardiography, cardiac computed tomography (CT), Positron Emission Tomography (PET) and

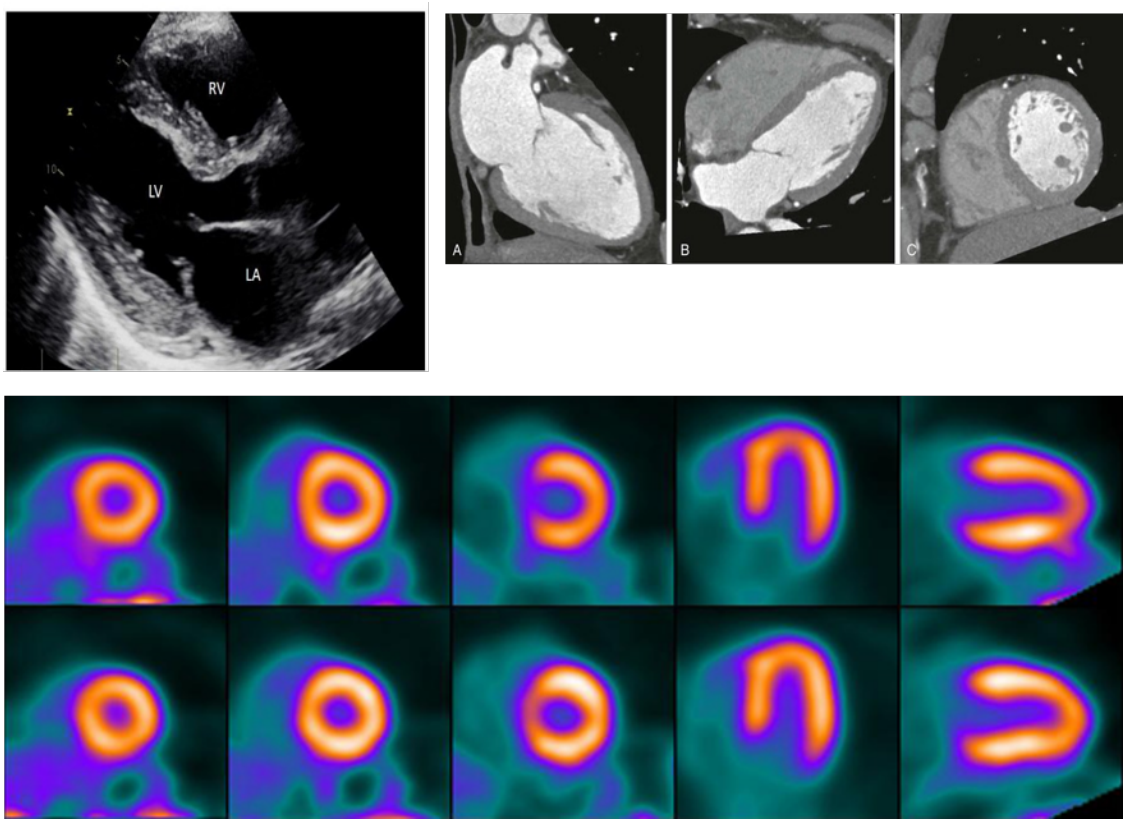


Figure 2.5: In clockwise direction, starting top left, examples of: Echocardiography, Cardiac CT and Cardiac PET. Figures adapted from [18], [19] and [20].

cardiac magnetic resonance (CMR). The first three will be introduced briefly, whilst CMR will be discussed in more detail, as the work in this thesis is primarily dealing with CMR data.

Echocardiography

Echocardiography is based on the use of ultrasound to create cardiac images. Ultrasound imaging uses the process of emitting pressure waves into the body where they spread out and get reflected off of tissue boundaries. The return time, also known as echo time, and the intensity of the reflections is recorded and used to create images. Echocardiography can be performed in 2D planar or 3D volumetric image acquisitions as well as utilizing the Doppler shift phenomenon to measure tissue and blood velocities. Hereby frequency shifts induced by moving tissue boundaries or objects provide measurements of velocities. This type of echocardiography is commonly used to assess blood flow and perfusion in the heart.

Echocardiography is easy and cheap to use, non-ionizing and portable. However, image quality is highly variable between users and the image has a low signal to noise ratio (SNR), as well as spatial resolution.

Cardiac CT

Cardiac CT uses x-ray beams to generate 3D images of the heart. It does so by beaming low-dose x-rays through the body at different angles and receiving them on the other side with x-ray detectors that measure beam attenuation. The amount of beam attenuation along the line of travel is indicative of the type and relative amount of tissue and bone that has been penetrated. The combined received signals from all angles form a so-called sinogram which can be used to re-create a depiction of the internal structure of the body using a method called backprojection. CTs are often used for coronary angiograms, whereby a contrast agent with high x-ray attenuation properties is injected and thereby facilitates a high contrast delineation of the coronary arteries. This is an effective method of analysing the levels of atherosclerosis. Whilst easy and cheap to use, CT uses ionizing radiation which can have harmful side effects on the body.

Cardiac PET

PET is a form of nuclear medicine and often used for functional cardiac imaging. It works by injecting a radioactive tracer substance into the body which decays by emitting positrons. These interact with nearby electrons and annihilate, producing gamma rays which exit the body and get collected by pairs of symmetrically placed gamma cameras. These signals are then used for image creation. PET tracers are typically paired with metabolic substances which are readily taken up by the myocardium, thereby providing detailed information about cardiac metabolism and perfusion. Drawbacks of PET imaging include the fact that it uses ionizing radiation, is costly to operate due to radiotracer manufacturing and has a low spatial resolution.

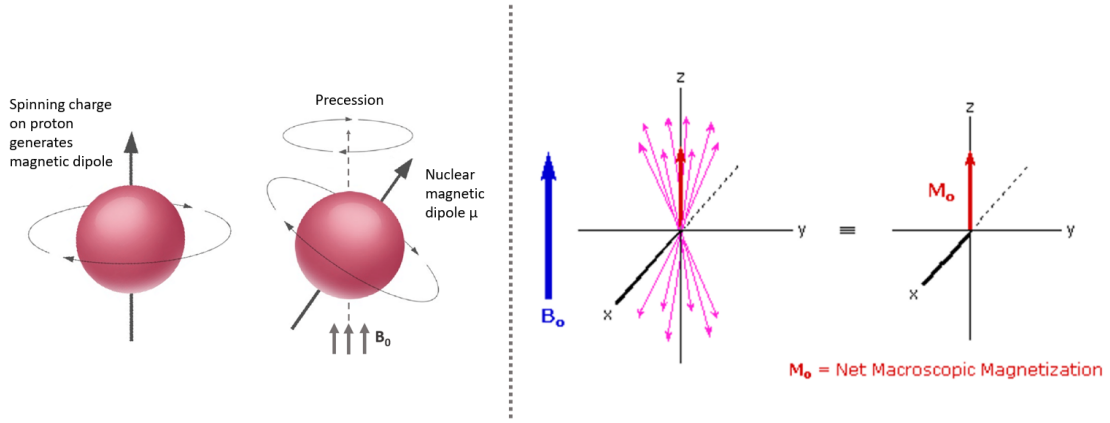


Figure 2.6: Nuclear Magnetic Resonance principle. Left: Proton precession, Right: Net magnetization. Figure adapted from [21] and [22]

2.2.2 MRI

MRI is a non-invasive and non-ionising biomedical imaging methodology that is widely used in clinical settings to visualise the internal structure and function of the human body. It finds applications in a range of tasks and organs and often forms the gold standard for radiological diagnosis. MRI is able to finely differentiate between tissues with high spatial and temporal resolution and offers a comparatively high soft tissue contrast. Its functionality is based on the principle of nuclear magnetic resonance (NMR), whereby magnetic fields are used to excite atoms in the body and change their nuclear spin. Changes in said property can then be measured and visualised to provide internal structural and functional information.

NMR Principle

NMR is based on the quantum property of nuclear spin which exists in both protons and neutrons. Whilst being a quantum characteristic in nature, it is often described in terms of classical mechanics to facilitate its understanding. As such, nuclear spin is commonly depicted as spheres rotating around an axis and consequently possessing angular momentum. As a result of being charged, magnetic moments are being created around the particles. The direction of moment vectors is opposite for protons and neutrons.

MR Signal

Hydrogen atoms are of particular interest in NMR for two reasons: firstly, their nuclei contain no neutrons, thereby making their net magnetic moment non-zero, and secondly, hydrogen is highly abundant in the human body. In its natural state, the magnetic moments of hydrogen atoms will be randomly oriented in the body, giving it a net zero overall magnetization. When placed in an MRI scanner, a large magnetic field (B_0) is applied to the body which causes all moments to align along its direction, resulting in an overall magnetization vector M_0. Usually the orientation of the B_0 field is chosen to run along the long-axis of the body, i.e. from head to toe. In addition to the alignment, the presence of the B_0 field also causes the moments to precess around the B_0 axis. The precession frequency, also known as Larmor frequency, is given by $w = yB_0$. Each precessing moment hereby consists of both a longitudinal magnetization component in the z direction, as well as a transverse component in the xy plane. When the B_0 field is first applied, the moments of the hydrogen atoms will precess out of phase with each other, thereby creating a net zero transverse M_{xy} magnetization.

A radio-frequency (RF) pulse is now emitted from a coil and applied at a right angle to the B_0 field with a frequency that matches the Larmor frequency. This creates a resonance in the moments, causing them to gain energy and flip away from the B_0 axis. At the same time they align in phase in the transverse plane, creating an M_{xy} magnetization vector. The angle between the overall induced magnetization vector and B_0 axis is termed the flip angle and depends on the magnitude and duration of the RF pulse. When the pulse is turned off several forms of relaxation occur. Spin-lattice relaxation refers to the gradual re-alignment with the B_0 field which causes the M_z magnetization vector to increase. It is defined by $M_z(t) = 1 - e^{t/T1}$ where T1 represents the time constant of relaxation. Spin-lattice relaxation depends on the difference between the natural frequency of the tissue and the Larmor frequency. The second relaxation is the spin-spin relaxation which refers to the gradual de-phasing of the individual moments and thereby a decrease in the M_{xy} vector according to $M_{xy}(t) = M_0 e^{-t/T2}$ where T2

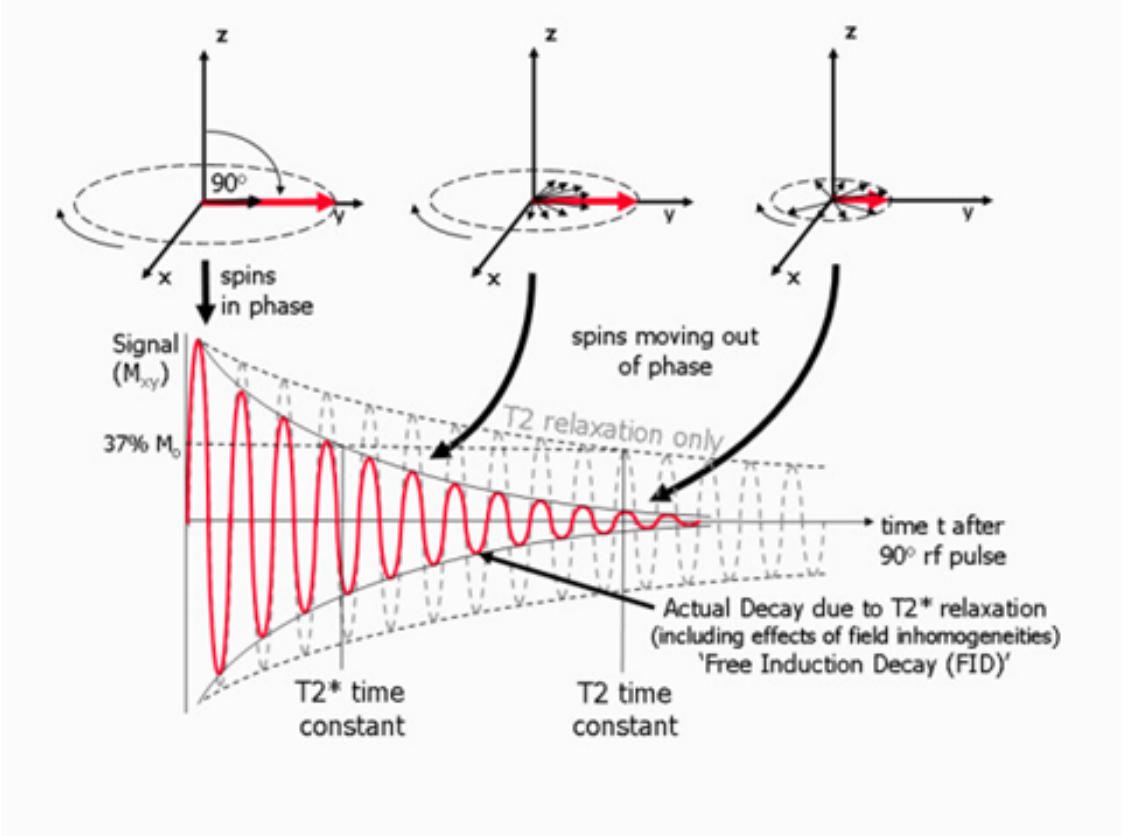


Figure 2.7: NMR Relaxation Process. Figure adapted from [23]

is the time constant of relaxation which again is an intrinsic tissue property. A third time constant, called T2* is often used which represents a modified T2 decay that accounts for field inhomogeneities and is shorter than T2. The change in the M_{xy} vector creates a time-varying RF pulse which is picked up by the coil, now acting as a receiver. This signal induces a current in the coil termed the Free Induction Decay (FID) which follows a decaying sinusoidal shape in line with a precessing and de-phasing transverse magnetisation.

Spatial Localisation

In order to decipher the spatial origin of the received signal, additional coils are placed around the body. These gradient coils apply linearly varying magnetic fields along three orthogonal axes to the body, thereby changing the local field strength and precession frequency. The induced RF pulse will now only resonate with moments in specific known locations where the Larmor frequency matches

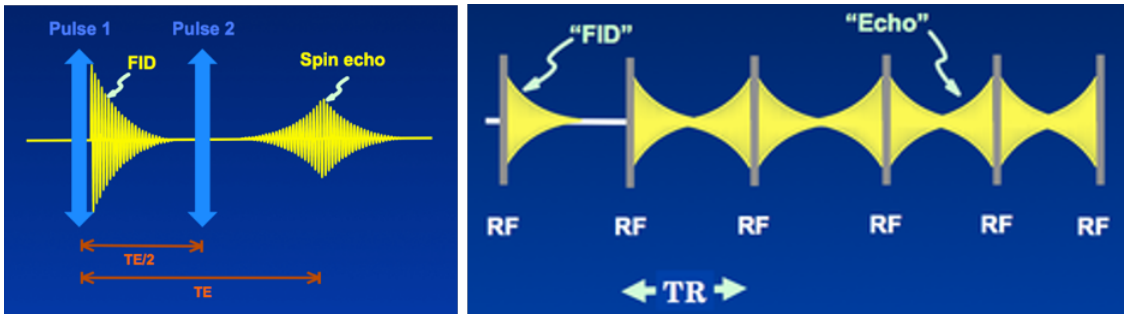


Figure 2.8: Left: Spin Echo Pulse Sequence. Right: Gradient Echo Pulse Sequence. Figure adapted from [24]

that of the RF pulse. The received image signals can thereby be attributed to these locations. The received signals exist in what is known as k-space and can be transformed into images by applying Fourier transforms.

Pulse Sequences

There are several pulse sequences that are commonly used in clinical MRI practice to acquire enough signal for image generation. Most of these consist of manipulations of repeated RF pulses which cause the de-phasing moments to re-phase and thereby produce M_{xy} magnetization and readable signal. The two most common groups of sequences are spin echo (SE) and gradient echo (GRE) sequences (on the left and right in Figure 2.8 respectively). In SE, the re-phasing is achieved by sending out a secondary RF pulse called the inversion pulse shortly after the initial pulse which causes the moments to flip 180°. As a result, they re-phase to create an "echo". The time between the initial RF pulse and the echo is known as echo time (TE), while the time between successive inversion pulses is known as pulse repetition time (TR).

MR image contrast depends on the combination of TE and TR values and on how much relaxation is allowed. Three main types of SE contrast images can be formed, namely T1-weighted, T2-weighted and proton density images. In T1-weighted images contrast primarily depends on T1 relaxation. Tissues with short T1 relaxation times, like fat, create high signal intensities. The TE and TR times are usually kept short to reduce the amount of T2 relaxation. T1-weighted imaging is highly suitable for showing anatomical details. T2-weighted images rely on T2

relaxation for contrast with tissues that have long T2 times, like water and most fluids, producing higher signal intensities than those with short ones. TR and TE times are longer in this form of imaging. Tissues with high water or fluid contents appear brighter which makes it suitable for perfusion imaging or MI localisation. Proton density imaging is the third main form which uses long TR and short TE to create a balanced T1 and T2 weighting. The signal intensity therefore depends more directly on the amount of signal from each location. GRE sequences use gradients to first accelerate the de-phasing of moments and then by reversing the gradient to re-phase the moments to produce an echo. This is typically much quicker than SE sequences and allows for more rapid imaging, however, it is also more vulnerable to field inhomogeneities.

2.2.3 CMR

Cardiac Magnetic Resonance (CMR) is an established form of generating information-rich data about cardiovascular structure and function. Having initially been established as a tool for studying myocardial metabolism in the 1970s [25], CMR has since developed into a broad set of clinical techniques that is now routinely used for the visualisation and analysis of the cardiovascular system. Due to its comparatively high soft tissue contrast and its lack of ionizing radiation, CMR is often superior to cardiac CT or echocardiography, making it the clinical gold standard for the diagnosis of a wide range of cardiovascular pathologies.

CMR Protocol

Typical CMR imaging protocols usually start by acquiring several scout images. These are low-dimensional images for localising the heart and determining its orientation. Aligning CMR imaging planes with the long axis of the body is usually not sensible, as these do not clearly transect the ventricles and atria. Instead, standard axes were defined that lie orthogonally to the long-axis of the LV that transects the apex and the centre of the mitral valve plane. Thereby the integrity of the chambers and the coronary arterial blood flow to the myocardium can be

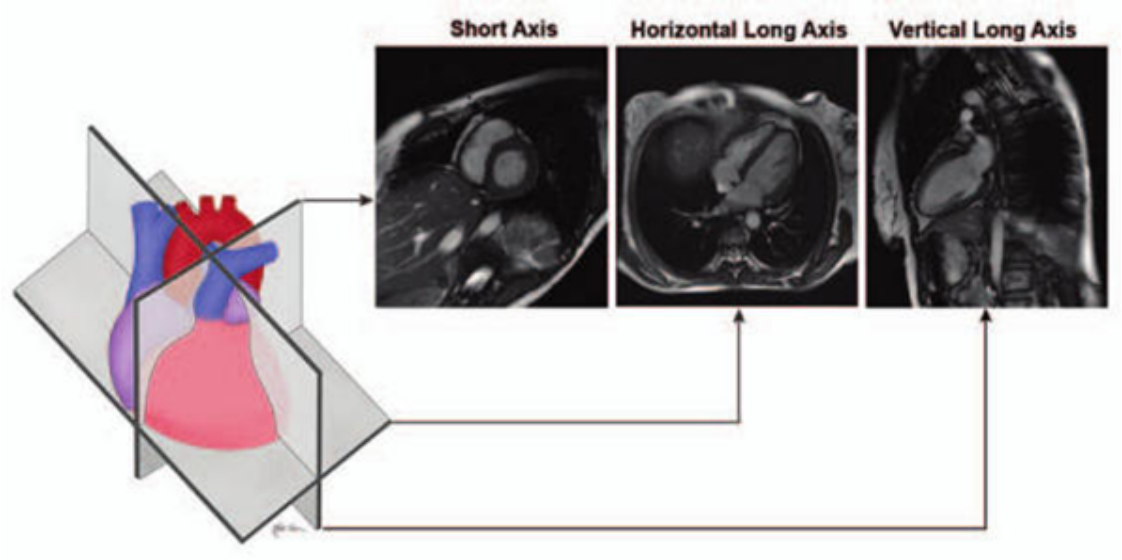


Figure 2.9: Cardiac Axes. Figure adapted from [26]

conserved which makes the understanding and analysis of the images much more straightforward. The planes perpendicular to these axes are termed vertical and horizontal long axis planes. The plane perpendicular to the long-axis (LAX) of the heart is called the short-axis (SAX) plane.

Cine

Following axis alignment, a stack of short-axis scans as well as the long-axis scans are taken which usually lasts 30-45 minutes. A commonly used pulse sequence for this is a GRE type sequence named balanced Steady-State Free Precession (b-SSFP) [24]. This type of sequence uses a train of back to back FID and echo pulses to generate rapidly repeating states of transverse magnetization and MR signal. It is particularly well suited for CMR imaging, since it offers short pulse repetition times resulting in high signal-to-noise ratios (SNR) as well as high contrast-to-noise ratios (CNR) [24]. It can also be both T1 and T2 weighted and is flow compensated. Potential problems arising with this sequence are banding artefacts, particularly at higher magnetic field strengths [27].

To get a better understanding of the functional behaviour of the heart, a method called cardiac cinematographic (cine) imaging is commonly applied which consists of imaging the entire cardiac cycle and stitching the frames together to form a

time sequence. This enables clinicians to analyse blood flow dynamics, myocardial perfusion, myocardial contraction and wall motion abnormalities. It also allows for the calculation of clinical parameters like ejection fraction. Usually a cine scan consists of 25 frames across the cycle.

ECG Gating

In order to time the acquisition of individual frames, scans are commonly synchronised with a patient's ECG in a method called ECG gating. Prospective gating identifies a trigger point, usually the R peak of the QRS complex, from which it calculates time intervals for the individual frames, taking into account the patient's heart rate. Retrospective gating continuously captures images and orders them afterwards based on their corresponding ECG signal. The final cine scan consists of frames acquired over several cycles. CMR resolution varies based on the imaging sequence used, however, SAX images usually have a spatial resolution of 1.5mm with an 8mm slice thickness and a 2mm gap between adjacent slices. Partial volume effects clearly play a significant role here.

CMR Problems

A major difficulty in cardiac imaging is patient breathing, as it disrupts the localisation of the heart and causes slice misalignments. To overcome this, imaging protocols are often run in breath-hold mode, whereby patients aim to maintain a constant lung volume for a short period of time following either an inhalation or an exhalation. This has the advantage of providing a higher spatial resolution scan. Patients of low functional status who are unable to hold their breath sufficiently long are scanned with free-breathing protocols, whereby belts are placed around their torso to measure diaphragm position and adjust for positional variability. Breathing is also commonly joined with ECG gating for better results.

In CMR acquisition time is a major issue. Long scans increase the effect of motion artifacts and blurring, however, they also allow for longer k-space sampling and higher spatial resolution. Field inhomogeneities and flow artifacts cause further issues. Lastly, pixel intensities in CMR scans are not absolute values,

contrary to Hounsfield units in CT, but depend on the magnetic field strength, the type of coil and the type of scanner. This can cause problems when trying to compare and combine scans.

Further CMR Methods

3D CMR scanning is becoming more feasible as pulse sequences improve and scanning time decreases. However, they are still rarely used as the effect of motion artifacts and the long scanning time are still an issue.

MR tagging is a technique by which short-axis planes of myocardial tissue are marked using RF pulses [28]. The resulting pattern of dark lines on the MR image moves synchronously with the underlying marked tissue and can hence be used for the visualization of cardiac dynamics, allowing for calculation of myocardial wall strain. By using the standard 17-segment model of the heart [29], strain curves can be calculated for each cardiac segment and areas of dynamic dysfunction can be identified. This approach is highly quantitative and allows for good objectivity in the assessment.

Late gadolinium enhancement (LGE) CMR is a method for measuring blood perfusion and works by injecting a gadolinium contrast agent into the patient's bloodstream prior to imaging. Myocardial blood perfusion and coronary artery stenosis can thereby be assessed. Due to the slow T1 relaxation of gadolinium, imaging at a later point in time also allows for myocardial infarction assessment, as regions with MI have longer blood retention times. This enables clinicians to assess the location and severity of myocardial infarctions and perform risk assessments for further infarctions.

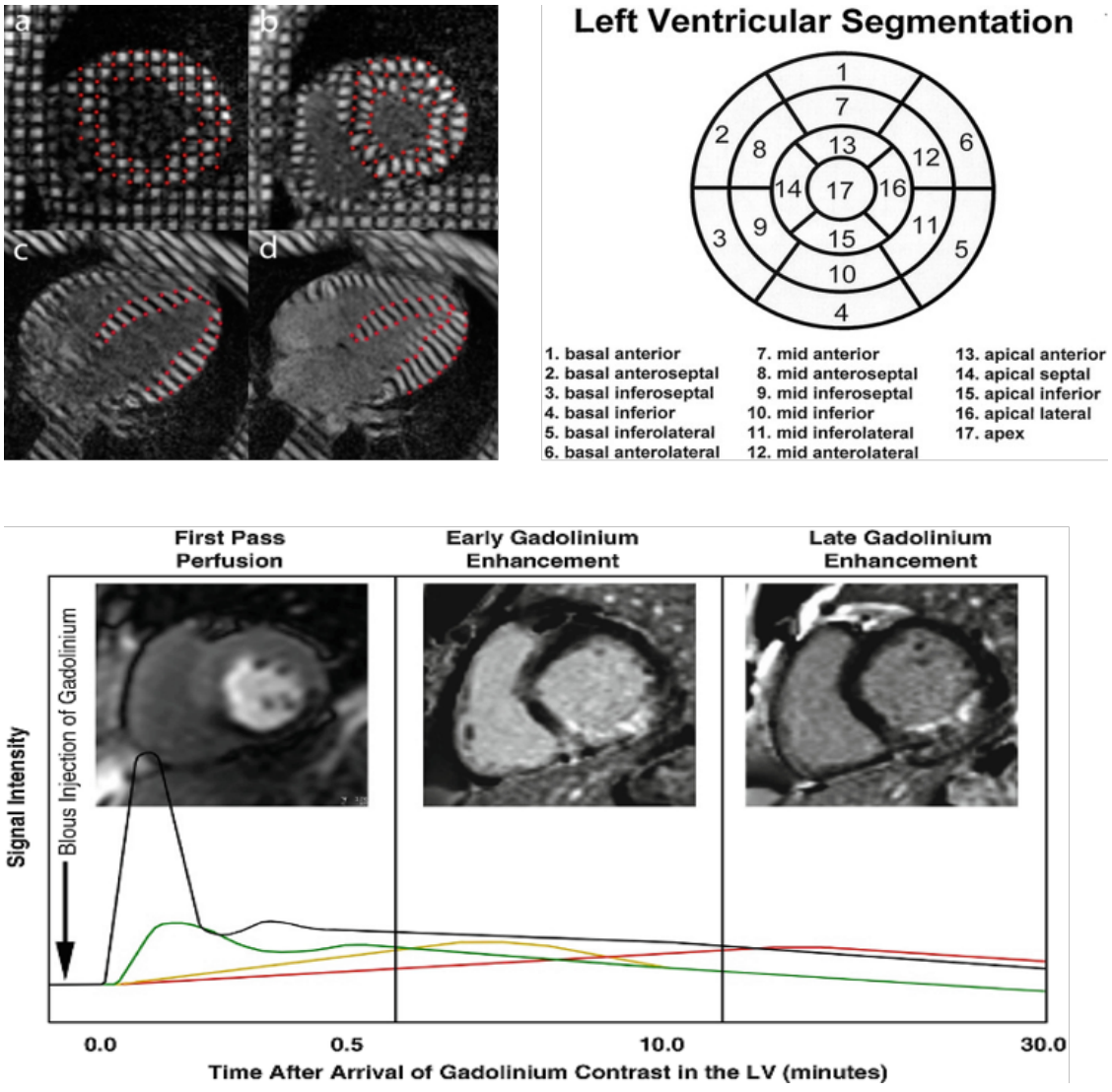


Figure 2.10: In clockwise direction starting top left: MR Tagging, 17-Segment Model and Myocardial Perfusion Imaging. Figure adapted from [30], [29] and [31].

2.3 Deep Learning

2.3.1 Introduction

The "universal approximation theory" states that all continuous functions can be approximated given a set of nonlinear activations and hidden layers with finite numbers of neurons. While theoretically true, the practical achievements of such "neural networks" (NNs) like Rosenblatt's 1957 perceptron [32] and that of various other models of the following decades [33] fell significantly short of such lofty expectations. The lack of computational power and the unavailability of large data sets, as was often the case in the latter half of the 20th century, simply did not allow for more and neural networks continued to be outclassed in performance by other techniques. While the rediscovery of the backpropagation algorithm [33] and developments like momentum, adaptable learning rates and max pooling in the 1990s represented significant advances in the field, it was not until the mid 2000s, when GPU-based implementations started to take off, that neural networks began to outperform more traditional machine learning approaches and began to live up to their theoretical promise.

In 2006, Geoffrey Hinton coined the term "deep learning" (DL) which has henceforth been used to refer to models which, based on underlying data sets and training regimes are able to extract complex patterns which enables them to produce state of the art results on classification, regression, or data generation tasks in the fields of natural language processing, computer vision and medical image analysis.

2.3.2 Neural Networks

Multilayer Perceptron Structure

In 2012, the performance of AlexNet [34] on the ImageNet data set presented a significant milestone for DL and displayed the potential of Convolutional Neural Networks (CNNs), a subset of NNs. The function of CNNs like AlexNet is largely based on that of the multilayer perceptron (MLP), an example of a feedforward neural network (FNN). The structure of an MLP can be seen in Figure 2.11.

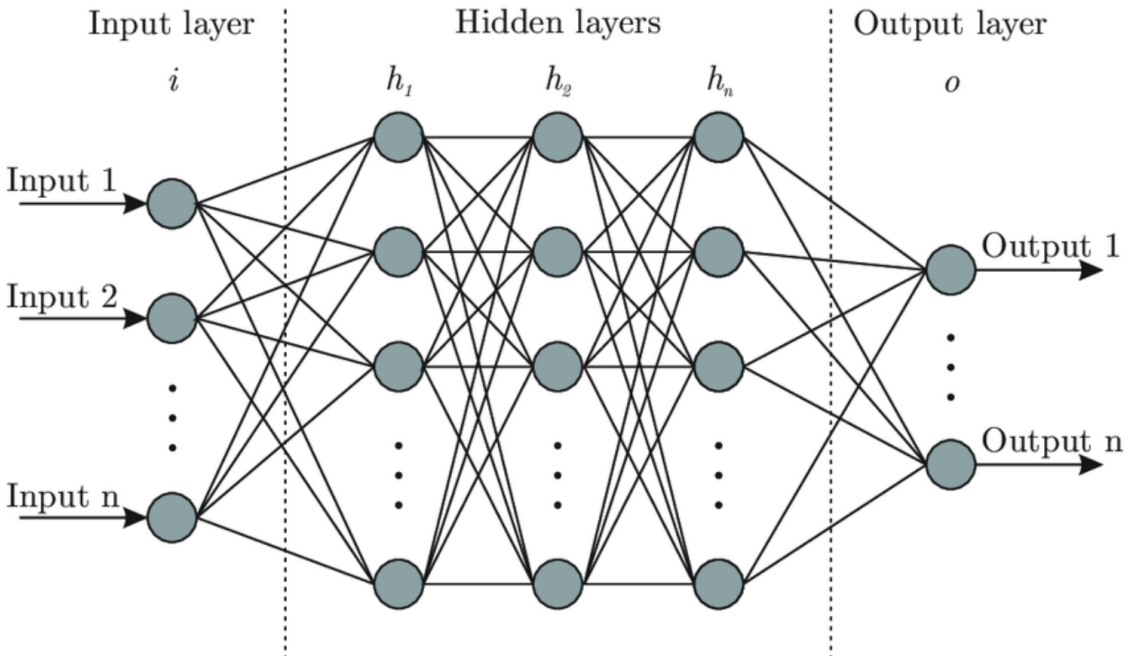


Figure 2.11: Simplified Neural Network Structure. Figure adapted from [35].

Taking a classification task as an example, the objective of the network is to find some function f that correctly maps inputs x to some label y. The mapping function of that network is therefore $y = f(x, \theta)$, where θ represents the parameters of the network that are learned and optimised in order to achieve the best approximation. The network usually consists of several layers of nodes where each node is connected to all or some nodes in the following layer. A network has an input layer, several hidden layers and an output layer. Outputs of one layer form the inputs to the next layer. Each node takes inputs from previous nodes, multiplies them by their connection weightings, adds a small bias term to prevent the node from "dying" during the training process, and then inputs them into a non-linear activation function to generate an output for that node according to the function: $Y = F(\Sigma(weight * input) + bias)$. Hence $y = f_1(f_2(...f_n(x)))$ where n is the number of layers. Rectified Linear Units (ReLU) are commonly used as the standard activation function for many applications. A ReLU activation performs a thresholding function on its input, setting negative inputs to zero and scaling positive inputs linearly, as seen in Figure 2.12. It has been shown to be superior to more traditional sigmoid or tanh functions primarily due to its ability to overcome the vanishing gradient

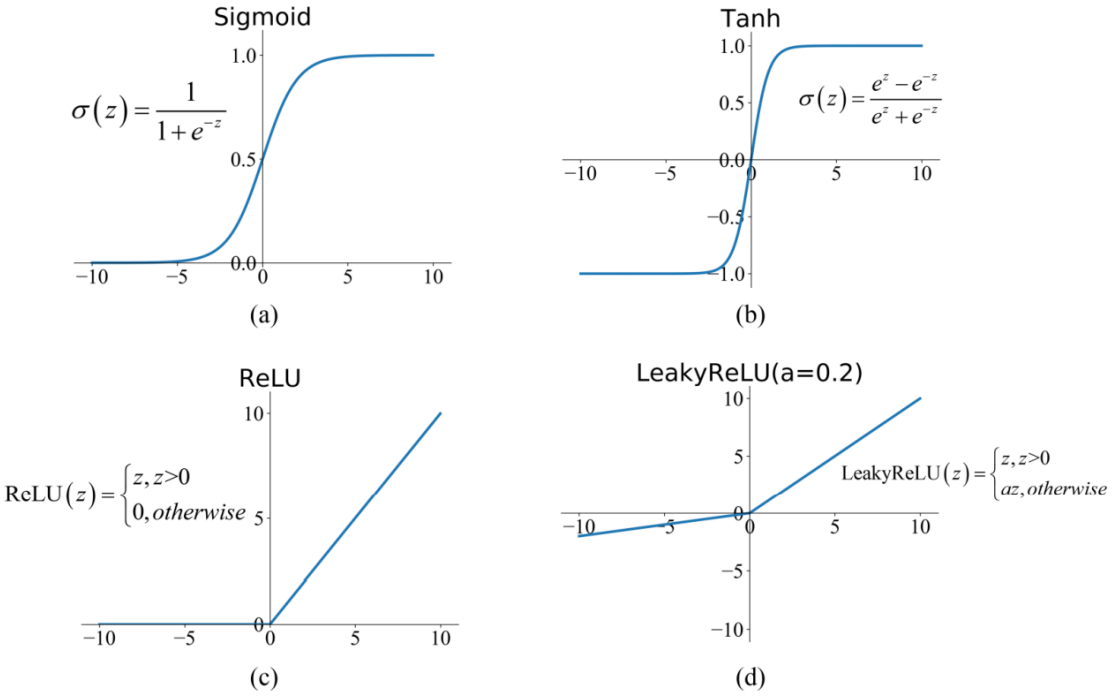


Figure 2.12: Examples of common neural network activation functions. Figure adapted from [37].

problem [36]. Variants of the ReLU, namely the ELU, SELU and leaky ReLU address problems of the ReLU such as the fact that all negative inputs get squashed to zero which can impede network performance.

MLP Training

NN training can be classified into supervised, unsupervised and semi-supervised, depending on the availability of training data labels. In a supervised training setup, ground truth labels are available for all training samples. As mentioned, the training objective of the network is to drive $f(x)$ to match the desired approximation $f^*(x)$. To achieve this, the network training procedure is composed of the following steps. Network inputs are propagated forward through the network resulting in an output prediction y. The prediction is compared to the actual label $\hat{y}$ of sample x and an error or loss is computed through a loss function. In the case of a binary classification task, binary cross entropy (BCE) would serve as a suitable loss function, as defined in equation 2.1, where N is the number of samples, $\hat{y}_i$ the target label value and y_i the output prediction.

$$BCE\ Loss = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i) \tag{2.1}$$

The objective of the training process is for this loss to be minimized. Since the loss is a function of the network parameters θ, in this case the weights of the connections between the nodes, the loss can be minimized by adjusting the weights accordingly. To do so, gradients of the loss with respect to the weights are computed. A step in the opposite direction of the steepest gradient on the multidimensional loss surface would result in a lower loss and hence a better network performance which is desirable. To do so, optimisation algorithms like Stochastic Gradient Descent (SGD) are implemented which take the average gradient of multiple data samples and adjust network weights accordingly. This step is called backpropagation and was first introduced by LeCun et al [38]. While the direction of the weight update step is given by the gradient, the size of the step is given by the learning rate and needs careful tuning for optimum performance. The training procedure is iterative whereby batches of training samples are sent through the network to produce an error from which gradients are computed. Starting at the final output layer, gradients are then updated sequentially by stepping backwards through the network layer by layer during backpropagation. This is repeated until the loss function has converged and an error minimum or performance maximum has been reached.

Improved optimisation functions such as Adaptive moment estimation (Adam), are used more commonly in practical applications. In contrast to SGD, Adam incorporates momentum and adaptive learning rates. Momentum is a concept designed by [39] to avoid local minima convergence. It works by modulating the learning rate in response to the agreement between the gradients of the previous update steps. The more these gradients align, the more the network can be confident that it is moving in the correct direction along the error surface. Hence the momentum term increases and so does the learning rate. Adaptive learning rates mean that the algorithm adapts the initial learning rate for each weight independently to allow for faster convergence. The number of nodes and layers in a network can vary depending on the task and often requires extensive hyperparameter

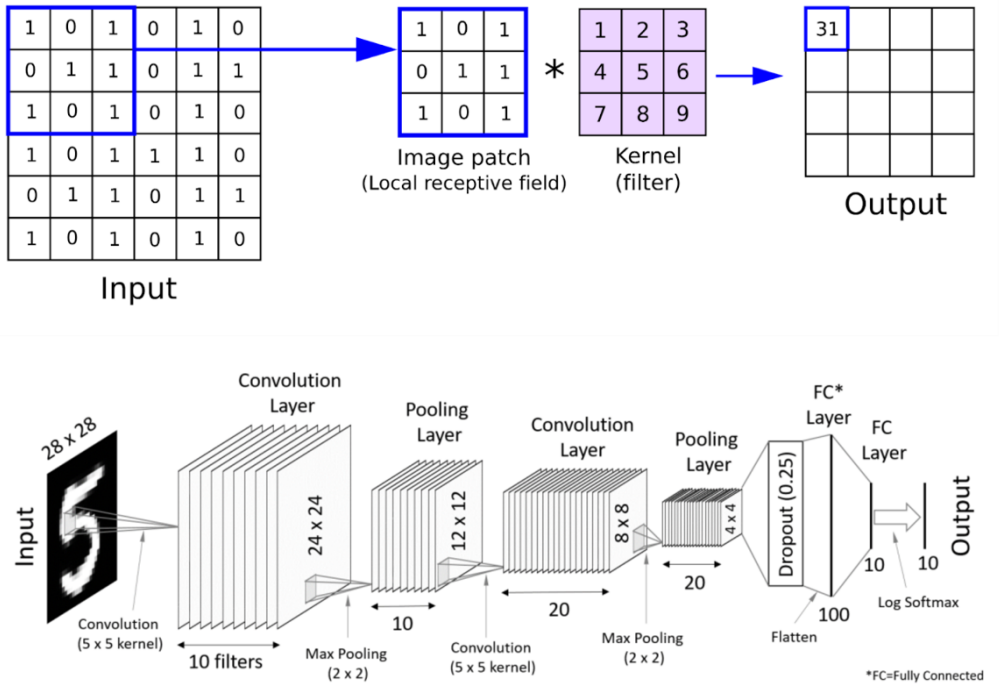


Figure 2.13: Top: Convolution operation. Bottom: Example CNN structure. Figure adapted from [40].

optimisation. Common problems that can occur include overfitting. The problem of overfitting refers to a situation when the network learns a function that is too closely aligned to a particular set of data points, effectively learning most instances. As a result the network fails to generalise well to unseen data sets, causing it to perform badly on prediction tasks. Overfitting occurs with increasing likelihood when the number of network parameters outweighs the number of training samples in a given task. Dropout is a common regularisation technique used to overcome overfitting whereby connections between nodes are randomly switched off during training time. This causes the training process to become noisy and forces nodes in a layer to co-adapt which in turn makes the model more robust. It also has the effect of encouraging the learning of a sparse representation due to its sparse activation of layer nodes.

2.3.3 Convolutional Neural Networks

CNN function

CNNs are an evolution of feedforward neural networks (FNNs) in the sense that they are designed specifically with 2-dimensional (image) data in mind. Whilst in FNNs independence between neighbouring data points can be assumed, this is not the case for CNNs, as image pixels are usually highly dependent on their local regions. Hence, instead of fully connected layers, CNNs primarily employ an alternating sequence of convolutional and pooling layers where the former utilise shared weights in the form of kernels that are convolved with regions of an input layer in a sliding-window approach to produce outputs. The weights of a kernel are the network's predominant learnt parameters. The same kernel is applied across the entirety of the input image, meaning the kernel weights are shared across all pixels. In comparison to a fully connected layer this represents a drastic reduction in learnt parameters. The convolution operation works by sliding the 2D kernel across the input image in strides of predefined length and convolving, e.g. multiplying and summing its weights with the underlying receptive field. Furthermore it is assumed that similar features can be found across the entire image space, i.e. feature position invariance exists. Each kernel produces an individual output feature map and usually multiple kernels are used per input layer resulting in several feature maps which are summed together to form the layer's output.

Following convolutional layers, pooling layers, like max-pooling, are used to downsample the input features. They are used to both reduce input dimensionality, but also to increase the effective receptive field of kernels in downstream convolutional layers and to introduce spatial invariance [41]. They also help against overfitting. As the layers progress, the network stitches together localised high-frequency information into more global features. Following a few convolutional and pooling layers, CNNs usually employ a few fully connected layers before any final classification or regression layers. The weights of the kernels are learnt in the same fashion as MLP learning occurs, through computing the gradients of the loss function, and updating network weights with optimisers during backpropagation. The CNNs

multi-layer hierarchical structure of decreasing image size and increasing receptive field allows the network to process information at various levels of resolution, and extract low, medium and high level features. Low level features, usually computed in the first few layers, include straight lines, edges, and curves which can be found across the image space. High-level features which are combinations of low and mid level features, often depict more abstract understandings of the input data.

CNN Developments

Since the introduction of AlexNet in 2012 and the start of "deep" networks, research in the field of CNNs has progressed by leaps and bounds. AlexNet outperformed the competition at the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012, primarily by adding more layers than was previously common. Instead of LeNet's [42] 5 layers, AlexNet used 8 layers. To overcome the added risk of overfitting that often comes with network depth, they introduced dropout into their network during training. Furthermore, to alleviate the vanishing gradient problem, ReLU activation functions were used instead of saturating sigmoid functions. It was also one of the first networks to use GPU power during training to boost performance. Shortcomings of AlexNet included the fact that it used rather large kernel sizes which resulted in aliasing effects [43].

To overcome the "black box" problem of CNN training which refers to the lack of understanding of the internal workings of the network, ZFNet [44] uses layer-wise visualisation to improve the understanding of feature extraction stages. This was done using deconvolutions which allowed them to visualise the activations of individual neurons in the network. This method was used to analyse the performance of AlexNet and discover that a lot of neurons were in fact inactive, prompting an adaption of kernel sizes and strides to smaller dimensions. Parameter tuning became an important step with the publication of ZFNet [43].

VGGNet [45] explored much more depth than previous networks with a total of 19 layers, allowing them to win the ILSVRC data set performance in 2014. Furthermore, they built upon the ZFNet analysis of smaller kernel sizes and

established 3×3 filters as the new norm. As an additional side effect, the smaller filter size reduced computational demands and allowed for even deeper networks. Furthermore, VGGNet introduced more modularity into architecture design, showing that simplicity and homogeneity of network architecture are key for good performance. For example, VGGNet only includes convolutions with 3×3 kernels and pooling operations with 2×2 kernels throughout the network.

GoogLeNet [46] published in 2015 was designed to utilise fewer computational resources whilst maintaining high accuracy at the 2014 ILSVRC competition. It achieved this by using its inception block architecture. Inputs to an inception block are branched to run through parallel convolutions at different scales before being concatenated at the output. The idea is that both coarse and fine information would be extracted at each level. To further reduce computational cost, global average pooling was used instead of fully connected layers at the end of the network. GoogLeNet also uses batch normalisation, a technique whereby data samples for each minibatch are standardised at the input of a layer. This ensures that the problem of internal covariate shifts is addressed [47].

Although initially the increasing depth of networks improved performance, as depth increased further, He et al. discovered the Degradation Problem [48] which describes the phenomenon of decreasing network training performance when adding more layers. This stems from an optimisation of the network towards local minima and is linked to the fact that it is usually easier for networks to learn a zero mapping $f(x) = 0$ rather than an identity mapping $f(x) = x$. To overcome this issue, He et al. [48] introduced residual learning with the ResNet. Instead of learning $H(x)$ (H being the function), one learns the residual mapping $F(x) = H(x) - x$, so that $H(x) = F(x) + x$. In effect this means connecting the input to the output which has shown to help reduce degradation as well as vanishing and exploding gradient issues. In theory any network depth is possible. In the worst case the residual is 0 and the layer does effectively not learn anything, but equally it does not degrade performance.

Wide residual networks [49] which rely on width rather than depth to improve network performance were introduced shortly after and showed improved perfor-

mance. Inception ResNet [50] and ResNeXt [51] were additional modifications of note of the original ResNet structure. SqueezeNet [52] followed on from AlexNet with the idea of making it faster and lighter. In fact they used 50x fewer parameters, yet achieved 3x faster training to achieve the same results as AlexNet.

DenseNet [53] addressed one of ResNet's problems, namely the fact that the identity connections across a ResNet block could cause layers to contribute little or no information to the learning process. DenseNet overcame this problem by using cross-layer connectivity which means connecting each preceding layer to the next layer in a feed-forward fashion, meaning feature maps of all previous layers were used as inputs to subsequent layers. By concatenating them instead of adding them, the network gained the ability to differentiate between preserved and added information in the network.

In the mid-late 2010s, focus shifted from parameter optimization to improved architectural design. Channel boosting [55] and feature map wise exploitation [56] are examples of such.

The concept of transfer learning [55] [57] has gained much traction as it enables faster and better learning for tasks with small data sets. Transfer learning is the idea that high-level features, i.e. edges, corners and basic geometric entities can be found in most imaging domains and therefore the first few layers of CNNs will extract similar features irrespective of the exact image domain. Instead of training a new network from scratch, Transfer Learning utilizes this fact by using a pre-trained network from a different task and freezing its first few layers, inhibiting further weight updates during backpropagation, and then training, or fine-tuning the network on the specific data set at hand. This essentially not only gives a "jump-start" to training performance, but also often leads to higher network performance if the pre-trained network has performed well.

Squeeze and excitation nets (SENetwork) [58] introduced the SE-block which suppresses less important feature maps but gives high weight to the class specifying feature maps. Ensemble learning [59] combines the predictions of multiple and

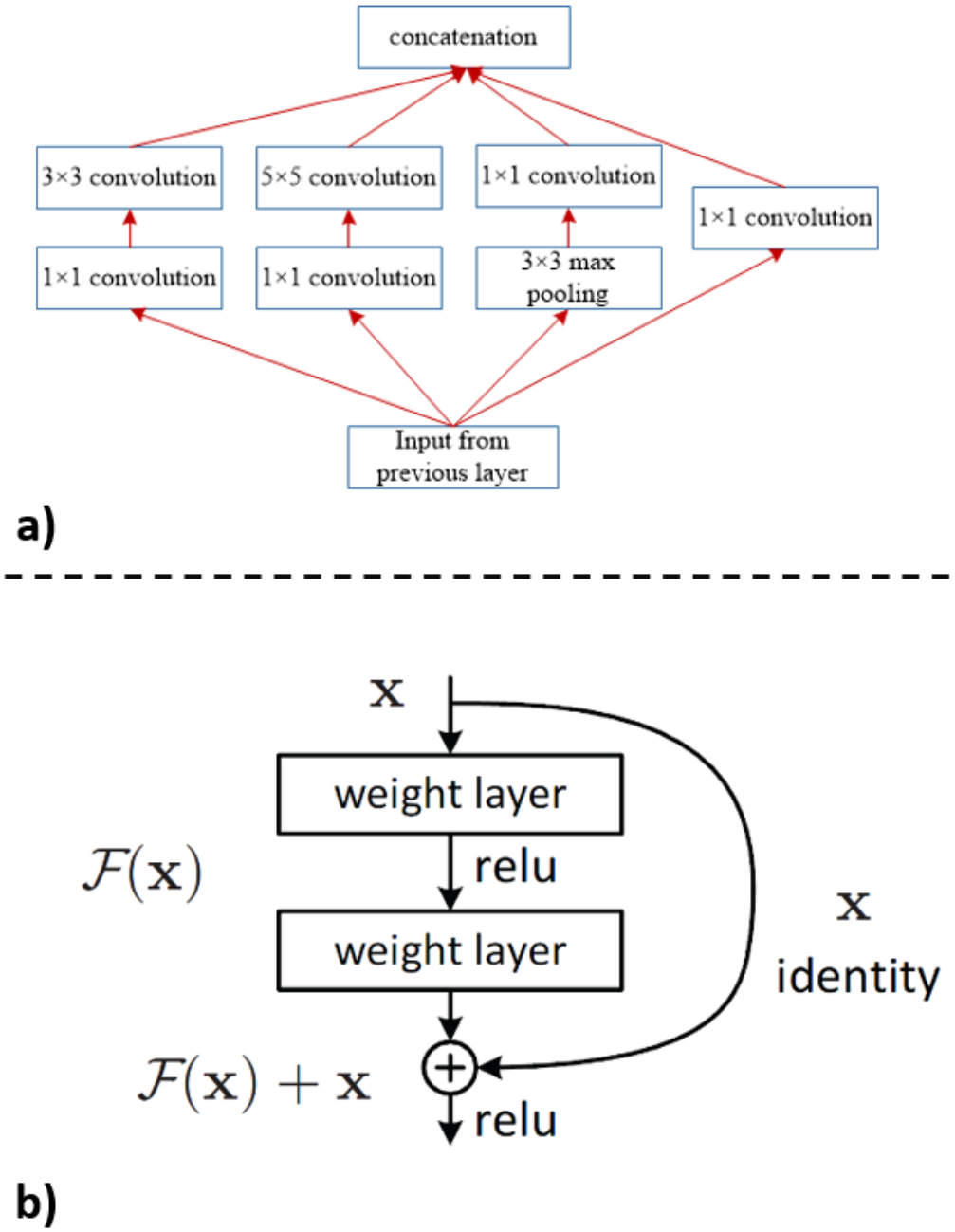


Figure 2.14: a) Depiction of the Inception Block structure. Figure adapted from [43]. b) Depiction of ResNet structure. Figure adapted from [48].

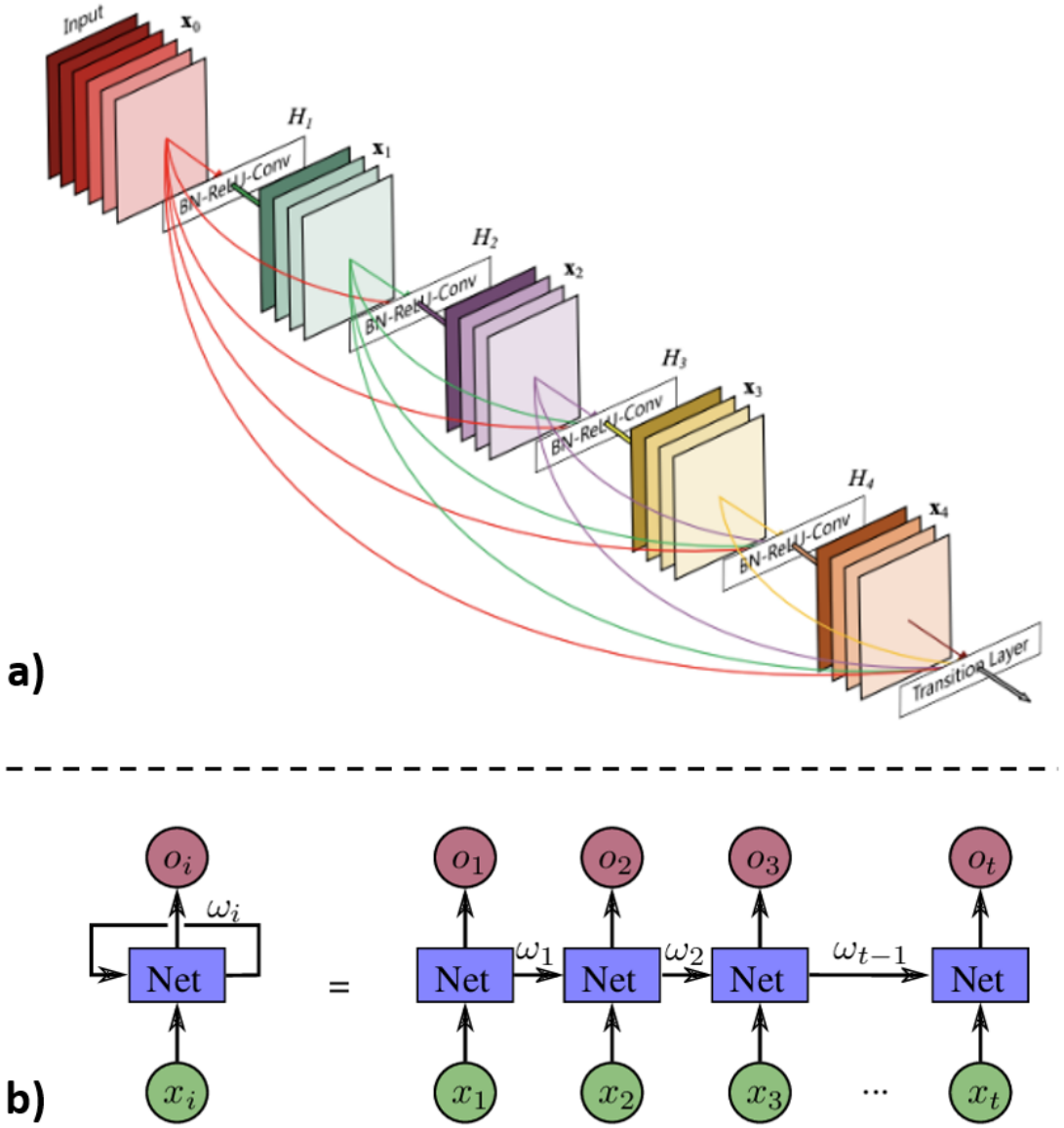


Figure 2.15: a) Depiction of the Recurrent NN structure. Figure adapted from [54]. b) Depiction of DenseNet structure. Figure adapted from [53].

diverse network architectures to reduce the variance of predictions and thereby reduce generalisation errors. It is seen as a important field for future CNN development [43].

Another region of future development is that of genetic algorithms to help with hyperparameter optimisation. This can be computationally infeasible to do in full parameter grid form, due to the multivariate nature of the problem. Genetic algorithms could offer a more feasible alternative. These algorithms are abstractly based on evolutionary processes [60] whereby possible solutions, or in this case

network hyperparameter settings, undergo cycles of mutations and recombination in order to produce new and optimised solutions. More specifically during each cycle, or "generation" of optimisation, each solution in the given population is encoded and evaluated based on a fitness function which can include criteria such as model loss, completion time and resource utilisation. The best performing solutions are then selected following a chosen stochastic method and recombination and mutation processes are performed to form a new population of solutions on which further evolution is based. The last step ensures population diversity and avoids premature convergence to less optimal solutions [60].

Recurrent Neural Networks (RNNs) are a separate class of neural networks focussing more on time-sequence modelling. Herbey nodes are not just connected to a preceding and succeeding node, but also to themselves. This allows them to accumulate memory over time. Long-short term memory (LSTM) and Gated Recurrent Units (GRUs) are improvements upon the original RNN structure and have achieved significant results. A big problem for LSTMs and RNNs in general is that when data sequences get too long, the performance of these networks drops. The reason is that the probability for keeping the context of a data sample that is far away from the current sample drops exponentially with distance. For example, in the case of a neural language sequence to sequence translation task, if the input sentences become really long, the memory of the first word is minuscule by the time the last word is reached.

As a solution, attention networks [56] were developed where the importance of words in a sentence in relation to a current word gives them different context weightings. Attention networks have enjoyed large success in natural language processing. The idea has also been merged with CNNs to form Transformer networks [61]. These allow for easier parallelisation than sequential RNNs, due to the CNN structure and completely eliminate sequential processing and recurrent connections. They rely entirely on the self-attention mechanism to capture global dependencies between input and output. Transformers have become the state of the art for NLP applications. They have also been increasingly applied in the image domain [62].

Graph Convolutional Networks (GCNs) [63] are another recent development in the DL space. Instead of using data that follows a structured Euclidean grid regularity, GCNs use data that follows non-regular graph structures. This makes them highly applicable to many bioinformatics tasks, but also useful for computer vision tasks where non-image data needs to be included.

Other interesting developments in DL in the past years include capsule nets [64]. Here, instead of being a scalar, the output of a node is a vector where the magnitude is the same as that of the equivalent CNN scalar, however the vector direction lies in a multi-dimensional space which represents features of the data. Information about object pose is captured which usually gets lost in maxpooling operations in CNNs. Capsule nets also employ a routing-by-agreement max-pooling procedure. This means that during the feedforward stage, iterative mechanisms between layers are used to determine connection weightings in order to route relevant information together. This allows them to form subsets of nodes which represent different properties of the data distribution and ensures that layers pass on information only to relevant neurons in the subsequent layers. The idea of capsules helps CNNs to be good at filtering out noise and retaining information about object pose and orientation.

Fully Convolutional Networks

Fully Convolutional Networks (FCN) [65] were introduced by Long et al. in 2014 for semantic pixel-wise segmentation. As shown in Figure 2.16, they consist of several stacked convolutional layers on a downsampling path, followed by an upsampling and prediction part. The output from each stage on the downsampling path gets upsampled to the original dimension using bilinear interpolation. The feature maps then get added and run through a final pixel-wise classification layer. In contrast to previous segmentation networks, FCNs do not use any fully connected dense layers, thereby reducing computational costs. They achieved state-of-the-art segmentation results on a number of challenges [65].

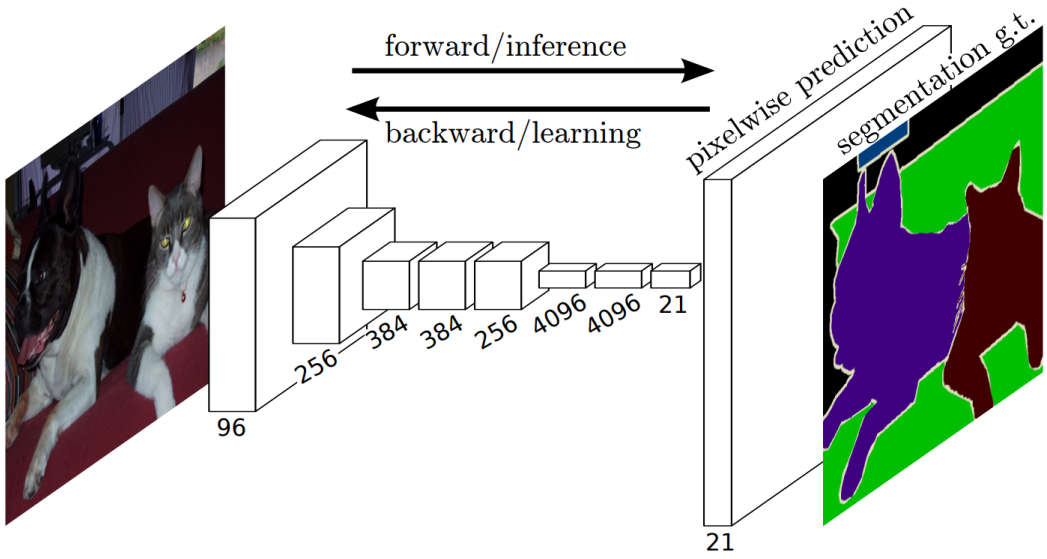


Figure 2.16: Fully Convolutional Network (FCN) Architecture. Figure adapted from [65].

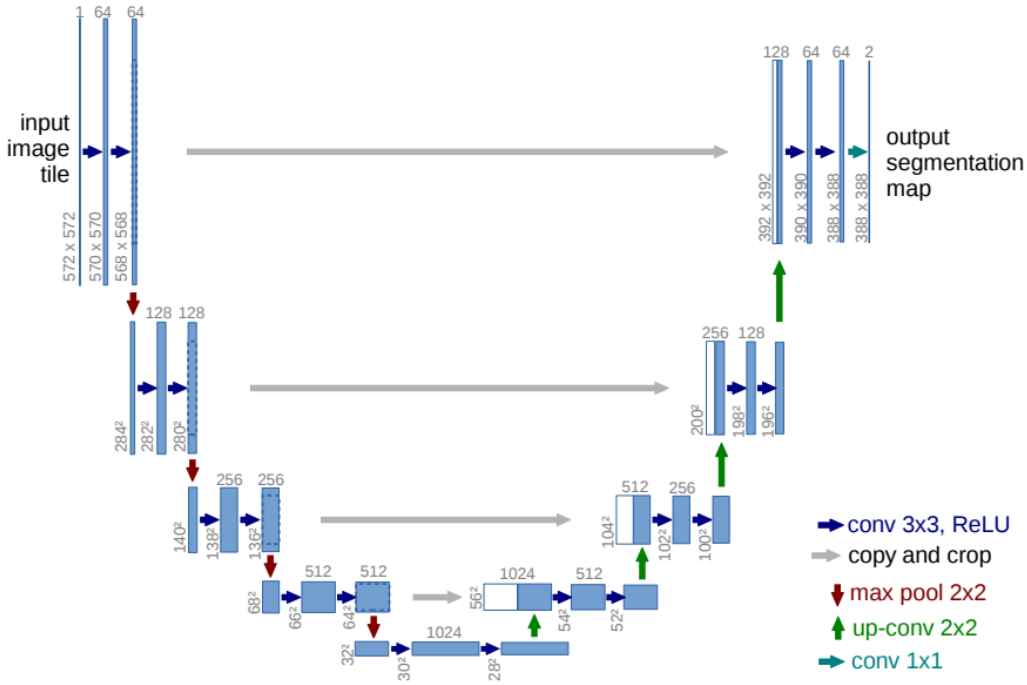


Figure 2.17: UNET Architecture. Figure adapted from [66].

UNET

The UNET [66] introduced by Ronneberger et al. at MICCAI 2015 was built upon the FCN and designed for segmentation tasks. It formed a significant breakthrough in CNN architecture research, particularly in the field of biomedical image analysis. The UNET consists of a downsampling path similar to the FCN, however, the upsampling process is performed in stages with further convolutions at each level of scale, giving it a more symmetric architecture [66]. Skip connections between the downsampling and upsampling branches are introduced to maintain high-frequency local information in the upsampling branch. In contrast to the FCN, these are concatenated, rather than added, before undergoing further convolutions. UNETs became the state-of-the-art architecture for segmentation tasks and have since become the backbone for further developments in the field. UNETs have been applied to super-resolution in [67] where they have achieved 2x peak signal-to-noise ratio (PSNR) compared to state-of-the-art by applying up to 20 weight layers. PSNR is defined as the logarithmic ratio between the maximum possible value of a signal and the power of distorting noise which is usually given as the mean squared error between the original image (i.e. ground truth) and compressed image and given in equation 2.2.

$$PSNR = 10 \cdot log_{10} \left(\frac{MAX^2}{MSE} \right) \tag{2.2}$$

Drozdzal et al. [68] have adapted UNET blocks by replacing the straightforward back-to-back convolutions with ResNet-like skip connections. Odena et al. have investigated the up-sampling process in UNETs and found that checkerboard like artefacts can occur [69]. They modify the deconvolution and replace it with a "resize-convolution" which causes these artifacts to disappear in a variety of contexts. The UNET has been expanded to deal with 3D data in [70] and [71] where they apply a Dice Loss as the network loss function and alter the UNET structure to compute the residual by adding upstream features to both the beginning and end of a layer

block. Dice loss is defined as $Dice\ Loss = 1 - Dice\ Score$, where the $Dice\ Score$ is given in equation 2.3 with A and B representing two sets of numbers.

$$Dice\ Score = \frac{2\,|A \cap B|}{|A| + |B|} \tag{2.3}$$

Oktay et al. [72] and Schlemper et al. [73] developed the Attention-gated UNET which includes attention modules to locate regions of interest on the downsampling branch which are important for the task at hand and then focuses on them during the upsampling process. These salient features help eliminate the need for further tissue localisation networks and help provide better insights into how the model makes predictions.

3D Networks

3D CNNs have been a more recent development in computer vision, due to the large data requirements that often come with it. Previously multi-view approaches for 3D object recognition yielded good results [74]. VoxNET [75] uses a 3D CNN to classify LIDAR point clouds. It converts point clouds into occupancy grids, similar to voxel grids, and then applies two layers of 3D convolutions followed by maxpooling and dense output connections. This was extended in [76], [77] and [78] which used object orientation as an additional loss. The penultimate dense layer of the network was connected to two multi-class classification layers, one representing the object class, the other representing categories of object orientation. This has shown to significantly improve object classification scores. It is reasoned that the network often does not learn the full underlying concept of pose variation and object classification but some representation that only partially generalizes to new data. The method presented in this work helps disentangle intra-class pose variations and interclass object classification.

In FusionNet [79] different networks are combined to achieve higher accuracy in 3D object recognition. One is a straightforward 3D CNN, the next is a 3D CNN with inception blocks and the third is a multi-view network that takes in stacks of 2D views. This approach has shown to yield better results than using each network

by itself. 3D networks used in medical volume classification are presented in [80] where CT volumes are classified into severity levels.

To address the computational requirements of 3D networks, [81] propose a quasi 3D convolution which applies 2D convolutional kernels separately on the three axes of 3D representations: the axial, coronal and sagittal orientations. Given that 2D convolutions are used, not only can pretrained 2D weights be applied, the network is also computationally less demanding than a 3D equivalent. The network has been shown to be superior to any 2D or 3D CNN without pre-training.

[82] combined a 3D version of the encoder part of the VNET, followed by a fully connected part and two separate losses to infer Tuberculosis Radiological Manifestations. Med3DNet [83] is a 3D transfer learning approach where data sets from several medical challenges are aggregated to form a multi-modality, organ and pathology data set on which a backbone can be trained which can then be used as a pre-trained model to boost other 3D tasks with insufficient training data. SagGAN [84] is a 3D VAE that learns the joint latent space distribution of the geometry and relationship between parts of 3D objects which are then disentangled in the decoder. This allows the user to separately control geometry and structure of generated objects.

2.4 Generative models

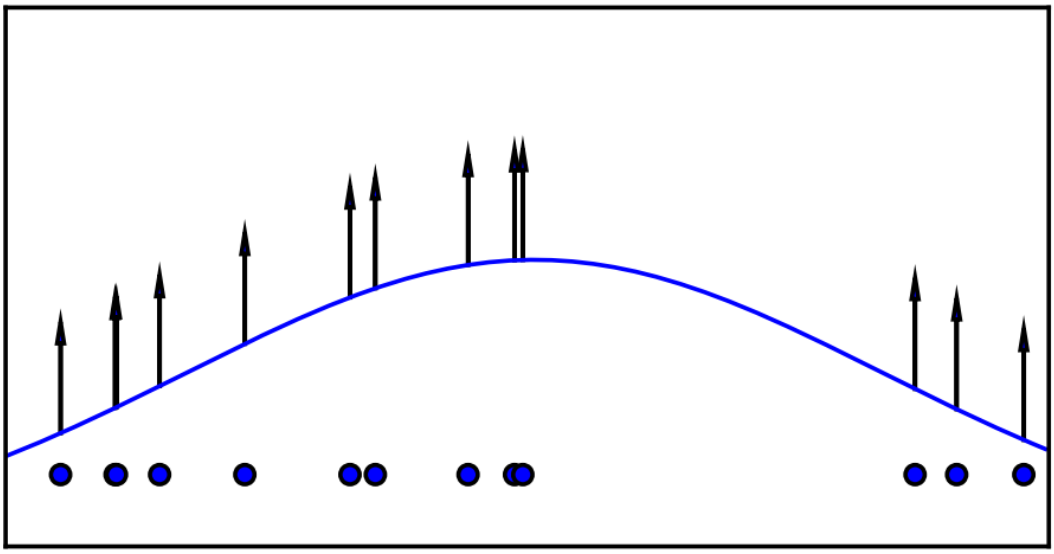
2.4.1 Introduction

In the field of deep learning, a distinction is commonly made between two classes of models: discriminative and generative. Discriminative models aim to model the conditional probability of a target label Y, given an observation x, i.e. $P(y|X = x)$. The modelling of a probability distribution, however, is not necessary for this class [85]. Generative models, on the other hand, model the joint distribution $P(X, Y)$. Their goal is to approximate the distribution P_{data} from which the training data was originally sampled as closely as possible. Having reached such an approximation P_g, new instances can be generated from this distribution which in themselves have no clear relationship to the training samples [86].

This allows one to gain the ability to manipulate high-dimensional probability distributions, which has a variety of applications. For example, in reinforcement learning it helps to plan subsequent steps given the current state. It can also be used for filling in missing data in semi-supervised learning, for multimodal output tasks and for the generation of realistic images through super-resolution [87].

Most generative models utilize the principle of maximum likelihood. Hereby a parameterized model is defined that provides an estimate of a probability distribution. The likelihood is the probability that the model assigns to the training data, i.e. that the data is sampled from the assumed distribution. The idea of maximum likelihood means choosing parameters of the model such that the likelihood is maximised [88]. This is equivalent to minimizing the KL-divergence between the data generating distribution (i.e. the training data set) and the model distribution. In log space we can sum rather than multiply over instances, which is why maximum log likelihood is more commonly used.

Generative models themselves are usually divided into two classes: implicit and explicit models. Explicit models use explicit parameterized log-likelihood functions to model the data [89]. Restricted Boltzman Machines (RBMs) [90], Deep Belief Networks (DBNs) [91] and Variational Autoencoders (VAEs) [92] are



$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{x \sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x} \mid \boldsymbol{\theta})$$

Figure 2.18: Graphical depiction of maximum likelihood process. Figure adapted from [88].

examples of such. VAEs will be discussed in more detail later on. The problem with explicit modelling is that the maximum likelihood estimation involves integrating over the entire parameter space of latent variables, which is often computationally intractable. As a result, approximation techniques are used. These, however, often do not generate satisfying results.

Rather than estimating the probability distribution explicitly, through approximation or not, generative adversarial networks (GANs), circumvent the problem of intractability. They do this by not trying to model any explicit distribution, but instead by generating data instances which are sampled from the distributions they aim to approximate [89]. Hence it can be said that GANs follow a likelihood-free inference approach [93].

2.4.2 GAN Introduction

Generative Adversarial Networks (GANs) were first published by Goodfellow et al. in 2014 [4]. They are a form of implicit density models which aim to capture

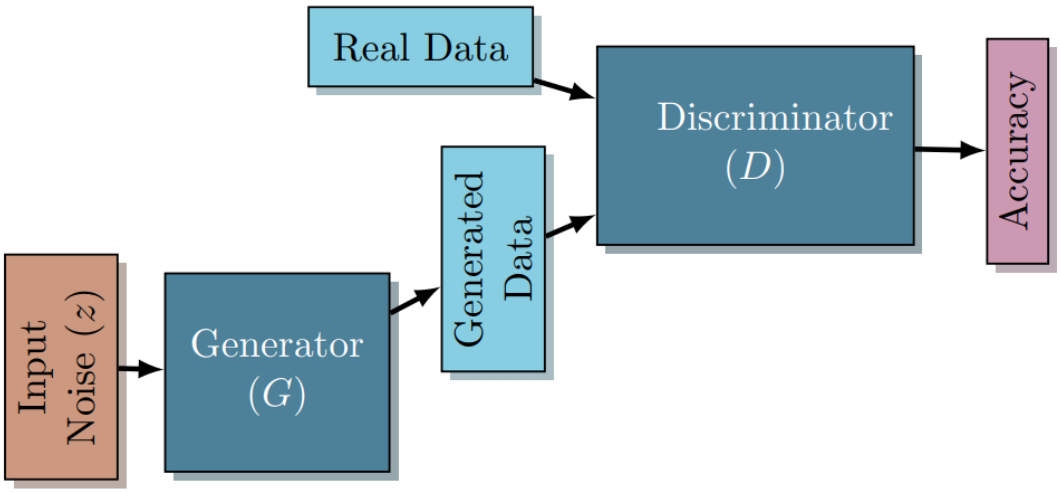


Figure 2.19: GAN structure. Figure adapted from [89]

the statistical distribution of data sets through a generative process. GANs fall under the category of unsupervised learning methods, since they learn hierarchical representations of data without the need for extensively annotated data sets. The idea behind GANs is loosely based on the Turing Test [94], whereby two agents compete in a zero-sum game. Here, one agent tries to fool the other which in turn tries not to get fooled. GANs work similarly in the sense that two neural networks are pitted against one another in an adversarial learning scheme. One network, the generator G, creates synthetic data samples which are fed into a second network, the discriminator D. The discriminator must then differentiate between fake, synthetic data samples and real data samples. Meanwhile, the generator aims to produce realistic looking samples to fool the discriminator. Through the optimisation of both networks, the generator learns to create more realistic data sets, thereby implicitly capturing an approximation of the real data distribution. A schematic of the GAN structure can be seen in Figure 2.19.

2.4.3 GAN Structure

The general structure of a basic GAN consists of two networks, the generator G and the discriminator D which can be modelled as any neural network with weights θ_G and θ_D respectively. In most cases, G has no access to real data, but purely learns

through its interaction with D. D on the other hand is provided with ground truth training data. The generator learns an implicit approximation of the statistical distribution of the training data during the training process. It maps an input from a noisy latent space z element of Z to the image space. In many cases the input is randomly sampled from a uniform distribution. More formally this is expressed as $G : G(z) \rightarrow R|x|$ where z is a sample from the latent space, x is an image and $||$ denotes the number of dimensions. The discriminator maps the image it receives to a probability score that the image belongs to the real data distribution in the form of a binary classification. More formally, $D : D(x) \rightarrow (0, 1)$.

The model updates its weights iteratively during training by minimising the objective function $J(\theta_G, \theta_D)$ through following the computed gradients ∇G and ∇D. Optimal training is achieved when $p_g = p_{data}$, where p_g is the learned distribution and p_{data} the real distribution. At this point the discriminator accuracy reaches a value of 0.5 meaning the generator is able to completely fool the discriminator into thinking the data samples are real.

2.4.4 GAN Loss

The training process of GANs consists of pushing the discriminator to maximise the probability of assigning the correct label to both real and generated samples. This means maximising $\mathbb{E}_{p_{data}}(x)[\log(D(x))]$, the probability of real data being classified as real and minimizing $\mathbb{E}_{p_g}(z)[\log(D(G(z)))]$, the probability of fake data being classified as real which in fact can be re-formulated as maximising $\mathbb{E}_{p_g}(z)[\log(1 - D(G(z)))]$. Equally the generator is pushed to minimize $\log(1 - D(G(z)))$, the probability of its samples being classified as fake. Effectively this means playing a minimax game between the generator and discriminator with the value function:

$$\min_G \max_D V(D, G) = \mathbb{E}_{p_{data}}(x)[\log D(x)] + \mathbb{E}_{p_g}(z)[\log(1 - D(G(z)))] \tag{2.4}$$

2.4.5 GAN Optimizer

The overall aim of generative methods is to achieve the point where $p_g = p_{data}$, i.e. the learned distribution equals or, in a more practical sense, closely approximates the training data distribution. This occurs at a saddle point of optimal training or otherwise known as Nash equilibrium. At this point a maximum and a minimum for the losses of the discriminator and generator respectively coincide. Goodfellow et al. show that reaching this point is theoretically possible [89]. A GAN is therefore less of an optimization problem where a local or global minimum is pursued, but rather a game. Training a GAN and approaching the saddle point is equivalent to approximately minimizing the Jensen-Shannon divergence (JSD) between the real data distribution and that created by the generator. The equation for JSD can be found in equation 2.5.

$$JS(P||Q) = \frac{1}{2}KL(P||P+Q) + \frac{1}{2}KL(Q||P+Q) \tag{2.5}$$

where KL is the Kullback-Leibler divergence:

$$KL(P||Q) = \int \left(P(x) \log \left(\frac{p(x)}{q(x)} \right) dx \right) \tag{2.6}$$

The JS divergence is a method for providing a measure of distance between two probability distributions. Therefore the minimization over Q means choosing the P_Q that is closest to the target distribution P_X in the JS divergence distance [95]. KL is a measure of distance in itself and minimizing it is equivalent to performing maximum likelihood estimations, as it is often used in explicit density models. The KL divergence, however, has shortcomings compared to the JSD, as it is famously asymmetric and possibly infinite when there are points such that $P_g(x) \to 0$ and $P_r(x) > 0$ [96]. This problem is called mode dropping or mode collapse and will be covered in more detail. Equally, for its reverse form, called reverse KL, there are points such that $P_g(x) > 0$ and $P_r(x) \to 0$, meaning unrealistic examples are generated [89]. JSD tries to cure that by extending KL (and reverse KL) divergence to calculate the symmetrical average of both as a distance measure of

one probability distribution from another. Some believe that one reason behind GANs' big success is switching the loss function from asymmetric KL divergence in traditional maximum-likelihood approach to symmetric JS divergence [97].

2.4.6 GAN Training

Theoretically convergence occurs at the global optimum when $P_g = P_r$. At this point, the optimal discriminator for a fixed generator is given by $D(x) = \frac{p_{data}(x)}{p_{data}(x)+p_g}$ For the optimal discriminator as given above, the generator is shown to minimize the following JSD [89]:

$$C(G) = -\log(4) + KL\left(p_d \middle\| \frac{p_d + p_g}{2}\right) + KL\left(p_g \middle\| \frac{p_d + p_g}{2}\right) \tag{2.7}$$

While the above equation has been argued by Goodfellow et al. to ensure convergence of the system, in practise it is not possible to train the discriminator to convergence at every iteration in finite time [89]. Hence that assumption is not entirely valid and causes problems of instability as will be discussed later.

Practically D and G are trained alternatingly, usually with i updates of D for every j updates of G. This follows the original proposal by Goodfellow et al. who suggest to use simultaneous SGD as the optimizer. Whilst training one network, the parameters of the other network are fixed. The values of i and j hereby are tunable hyperparameters. Arguments can be made for having $i < j$, as well as $i > j$, i.e. training one network more than once for every update of the other network. The former is argued to be useful due to the risk of discriminator saturation which can result in vanishing gradients for generator training. This is discussed further later on. The latter case, i.e. $i > j$, is argued to be useful since the generator gradient ultimately comes from the performance of the discriminator, hence a well performing discriminator is also desirable. Generally Goodfellow et al. suggest to not inhibit the power of the discriminator [88] but instead to use parameterization of the game where the gradient does not vanish.

2.4.7 GAN Problems

Problem of Convergence

Training a GAN usually comes with several difficulties. The first is that of instability and non-convergence. In theory GANs are designed to reach a Nash equilibrium at which each player cannot reduce their cost without changing the other players' parameters [98]. At this point the divergence between the target distribution and the generator's distribution should be at a minimum. However, GANs operate in a zero-sum game where improvements to one network come at a cost in performance of the other. As the parameters of one network are updated, the nature of the optimisation problem at hand changes, making the training process dynamic [88].

In practise this means that reaching convergence in finite iterations is often not possible. The above equation which has been proven to reach the saddle point assumes that we can train the discriminator to optimality which in practise is not the case given finite resources and capacity. Hence achieving convergence in GAN training can be an issue.

Additionally, the generator and discriminator losses obtained during training are in themselves not indicative of convergence. They can be smooth or strongly oscillating, either behaviour not being a straightforward indicator for network performance. This poses a big problem for evaluating GAN training [89].

Having said that, GAN training can be viewed as approaching Nash equilibria via trajectories that do not necessarily minimize a specific divergence at each step [98]. Furthermore it has been shown that optimum GAN performance does not always occur when a Nash equilibrium is reached, but rather a proximal equilibrium which captures the sequential nature of GANs, in which the generator moves first followed by the discriminator [99]. Lastly, what is considered optimum for a specific application might not in fact occur at the saddle point.

Various regularisation techniques have been implemented to improve convergence [100], [101], [102].

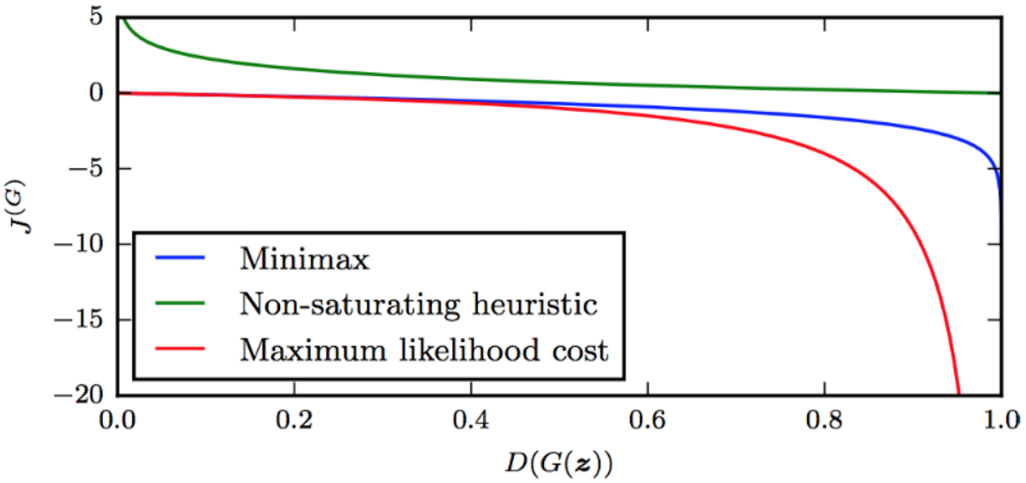


Figure 2.20: Curve shows generator gradient for different values of discriminator performance on fake inputs for different cost functions and highlights their problems. Minimax game (original GAN cost function where $J(G) = -J(D)$) and Maximum Likelihood (which assumes an optimal discriminator) both suffer from vanishing gradients when D is initially good at identifying fake samples. The latter also receives almost all gradient updates from the right end meaning some batches will dominate training. The non-saturating heuristic method trains G to maximise $\log(D(G(z)))$ instead of minimizing $\log(1 - D(G(z)))$ which provides much larger gradients in earlier training, however, it can have unstable numerical gradients for G [103]. Figure adapted from [88].

Problems of Vanishing Gradients

Both P_g and P_d lie on low-dimensional manifolds, meaning the intersection between them can often be small. This means that if the discriminator is allowed to train to convergence, i.e. the point at which the loss approaches zero and all labels are predicted correctly, it can result in a vanishing gradient for the generator, causing it to stop learning effectively.

Goodfellow et al. [4] argue that the problem of vanishing gradients is particularly present during early training when the discriminator will perform well, given that the generator's output is still easily distinguishable from real data samples. This can cause problems to the generator's training. As a result, it was proposed to use $max_\phi \log(D_\theta(G_\phi(z)))$ instead of equation 2.10 [4] which gives higher gradients for the generator when $D(G(z))$ is low. It can, however, also cause unstable numerical gradients for G [103].

Fundamentally to overcome the vanishing gradient problem, the loss function

requires alterations. The Wasserstein Loss [96] is such an example. The idea here is to replace the JS distance with the L1-Wasserstein distance. The reason why this is deemed more suitable is because it respects the geometry of the underlying space and also because it can compute the distance between distributions even when they do not intersect. A non-negligible intersection is quite common to occur in high dimensional applications when the target distribution lies in a low dimensional manifold. Whilst in more traditional generative methods this problem has been overcome to some degree by adding high-dimensional noise, it has the detriment of decreasing the quality of the likelihood estimation. Wasserstein Loss does not have such problems. In practical terms this means that the discriminator can be trained to optimality before training the generator, without introducing vanishing gradients [95].

Problem of Mode Collapse

Another common failure mode for GANs is that of mode collapse, where the generator tends to generate the same set of outputs, or modes, repeatedly for different inputs without effective learning. If the generator finds a set of outputs that work well on the discriminator, and the discriminator gets stuck in a local minimum, then it can happen that the generator just iterates between the well working sets of outcomes.

A potential reason for mode collapse is that the simultaneous SGD does not differentiate between solving the min-max or max-min problem. If the minimization is performed first, i.e $min_\theta \, \mathbb{E}[\log(1 - D_w(G_\theta(z)))]$, then the optimal generator would map all data points to where the discriminator assigns high values [95]. Alternatively it has been suggested that mode collapse can be related to having small dimensions in the generator input latent space, i.e having a low dimensional manifold.

It was suggested to use minibatch features [104] to address mode collapse. Hereby the discriminator compares an input to a minibatch of generated samples and a minibatch of real samples in latent space and uses the distance to judge whether it is too similar to other generated samples.

2.4.8 Improvements to GAN Training

It has been suggested to minimize the number of fully-connected layers in order to increase the feasibility of training deeper models. Further heuristic improvements include feature matching whereby the generator is trained to match the discriminator's expected intermediate features of its fake samples with the equivalent features of its real samples [104]. Next, minibatch discrimination helps as an extra discriminator input that encodes the distance between the sample in a batch and all other samples. This is intended to prevent generator collapse. One sided label-smoothing is another technique whereby ground truth labels for real data samples in the discriminator are changed to values <1 in order to reduce vulnerability of GANs to adversarial attacks [104]. Denton et al. [105] first suggested training with labels to improve subjective quality of outputs. Salimans et al. [104] confirmed that even if no explicit class information is incorporated in the generator, training with labels can still improve the quality. Heuristic averaging [104] is used to penalise network parameters if they deviate from a running average of previous results, similar to momentum. Other improvements to GAN performance through adapting the loss function include the Least Squares GAN [106] which uses a least square loss instead of the BCE loss proposed originally.

2.4.9 GAN Performance Metrics

As a form of unsupervised learning, the lack of ground truth training labels makes the validation of GANs non-trivial. Furthermore, quantitative evaluation is difficult as there is no explicit density estimation. This means that density distance methods such as JSD or KL divergence cannot be used, leaving only sample-based metrics. Nonetheless, a combination of qualitative and quantitative metrics have been developed and are used in practice. Qualitatively, human annotation is often used as a performance metric. Hereby human "experts" judge the realness of generated data. This can take the form of Nearest Neighbour comparisons [107], rapid scene categorization or rating and preference judgement. Quantitatively a wide range of surrogate tasks have been devised to generate metrics for assessing

GAN performance. Most commonly used metrics are the Inception Score (IS) [104] and Frechet Inception Distance (FID) [108].

$$IS(P_g) = e^{\mathbb{E}_{x\sim P_g}[KL(p_M(y|x)||p_M(y))]} \tag{2.8}$$

The IS, as described in equation 2.8 where M is the image classification model, is solely designed for evaluating the performance of networks that deal with natural image data sets, or data sets similar to ImageNet. It works by analysing the confidence of the InceptionV3 network [109] in classifying generated images into one of the ImageNet [110] classes after they have been passed through the last classification layer of the InceptionV3 net. The more confident the network is in its classification into a single particular class, the higher the Inception Score. This is, because it is assumed that for an image to be clearly classified into a particular class it must share many common features with said class. Given that the images in ImageNet are all real natural images, this means that there is a high likelihood that the generated image is also realistic looking. For natural image tasks, the IS has been shown to have a reasonable correlation with human judgement of image quality [104]. However, it has limited transferability to non-natural image tasks as it only evaluates the image generation model P_g, rather than its similarity to the data distribution P_r.

The Frechet Inception Distance (FID) measures the similarity between generated distributions and real image distributions within the same domain that were used to train the generator. The metric measures the Frechet distance between two multidimensional distributions of features gathered from the penultimate layer of the InceptionV3 network. The features are created by feeding in images sampled from the real and fake data distributions of the GAN respectively.

$$FID(P_r, P_g) = ||\mu_r - \mu_g|| + Tr(C_r + C_g - 2(C_rC_g)^{\frac{1}{2}}) \tag{2.9}$$

Where μ, C are the mean and covariances of the distribution of features for the real and generated data respectively.

While the InceptionV3 network is trained on natural images from the ImageNet data base, the FID metric can also be applied to non-natural image domains, given

that, contrary to the IS, it compares the performance of two external distributions instead of using the ImageNet distribution as the benchmark.

Other quantitative methods include Wu et al.'s 2016 method [111] to estimate log-likelihoods of generative networks by assuming a Gaussian observation model with a fixed variance and sampling from that. Bounliphone et al. [112] use Maximum Mean Discrepancies (MMDs) for selecting generative models and Lopez-Paz & Oquab [113] use the classifier two-sample test. These are, however, not necessarily model agnostic. Other model agnostic metrics are the 1-Nearest Neighbour classifier, Kernel Maximum Mean Discrepancy, Mode Score and Wasserstein distance [114].

Lastly application specific performance metrics can be applied on top of the aforementioned methods. In the context of CMR data for example, image segmentation networks can be used to extract domain specific features such as myocardial mass or ventricular volume. Distributions of such values can be compared between real and synthetic data sets to assess GAN performance.

As a more practical point, GAN models and weights must be saved frequently during training so that at inference time, these models can be loaded and new instances can be created from which performance metrics can be computed.

2.4.10 GAN Developments

Conditional GANs

Conditional GANs (cGANs) are one of the first major developments of the traditional GAN. Hereby both the generator and discriminator are class-conditional [115]. This means that the generative process can be guided, based on label information or other auxiliary information provided [116]. The objective function changes to:

$$\min_G \max_D V(D,G) = \mathbb{E}_{p_{data}}(x)[logD(x|y)] + \mathbb{E}_{p_g}(z)[log(1 - D(G(z|y)))] \quad (2.10)$$

This type of network can be used for image-to-image translation applications [116]. Furthermore, cGANs have been applied for style transfer [117], image inpainting tasks [118], face generation and face aging tasks [103].

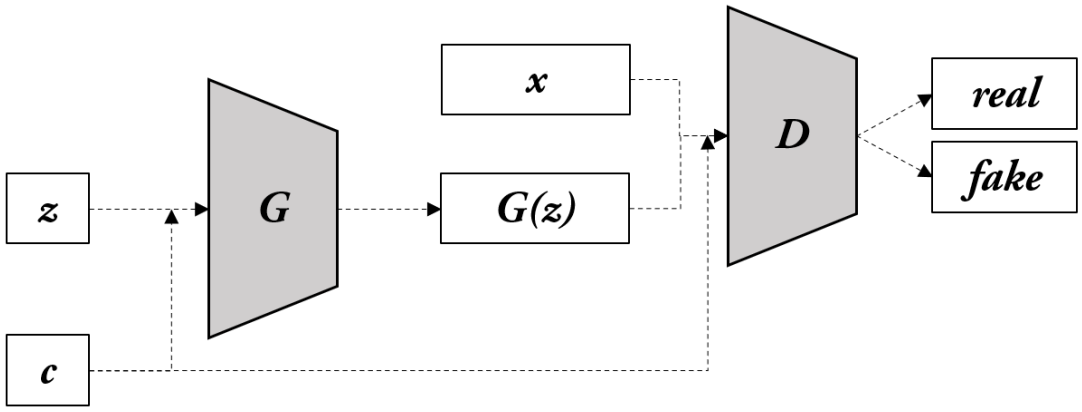


Figure 2.21: cGAN Structure. z is the latent generator input, c the conditional input and x a real sample.

The earliest GANs consisted of Fully Connected NNs, but were quickly taken over by deep convolutional GANs (DCGAN) [119]. These merged the convolutional approach of CNNs with the adversarial idea of GANs to create a GAN variant with significant advantages for dealing with image data compared to original fully connected GANs. DCGANs made use of global average pooling, batch normalisation and fractional strides in the generator for up-convolutions, all to increase training stability [120]. It was also shown that the GAN latent space is relevant and that arithmetic operations can lead to a clear visual impact in the generated images. For example a "smile" vector can be increased, causing the generated image of a person to adopt a smile [121].

LAPGANs [105] decomposed the generation process into multiple scales by using a cascaded network to form a Laplacian pyramid. At each layer, successive improvements were made to the generated samples, from rough to fine features. It showed better performance on standard image sets than the traditional GAN.

Previously mentioned LAPGANs and DCGANs are purely generative models, i.e. they map a latent space to the data space. However, it becomes tricky when trying to extract the learned data distribution and map a data space sample back to the latent space. This inference mechanism [87], is addressed by the ALI network [121] and Bidirectional GANs [122]. These provide an inference

network where the discriminator is trained on joint data and latent pairs. The generator not only includes a decoder component, but additionally trains an encoder going in the reverse direction. The encoders can be used for high level semantic understanding of the data.

Investigating the loss function of the GAN, Nowozin et al. [123] showed that f-divergences which are the more general case of KL divergence can be used as objective functions in the GAN setup. They propose the Variational Divergence Minimization method for estimating the parameters of the generator and simplify the saddle-point optimization procedure of the original GAN.

InfoGANs [124] introduces the concept of mutual information in order to interpret disentangled representations. Similar to cGANs, InfoGANs are able to generate conditional outputs. The categorical features, however, are determined by the network itself in an unsupervised fashion. It works by having an additional input vector to the generator in addition to the usual noise vector z. This vector c, called latent code, targets significant structured semantic features [103]. The goal of InfoGAN is to maximise the mutual information between the generator distribution and the latent code, in order to force c to contain meaningful features of the data [120], such as the direction of lighting on an image of a face [88].

Auxiliary Classifier GANs (ACGANs) [125] are similar to cGANs and InfoGANs in that they add conditional information to the generator input, though for ACGANs this information is restricted to just the class label. ACGANs vary from the other two in that they also train the discriminator to predict the class label from the image. This has shown to result in higher resolution output. They also first made use of the Inception network for performance validation.

Stacked GANs [126] use a set of stacked conditional GANs to create photo realistic image synthesis. The network is made up of two stages. The first is a conditional GAN that generates low-resolution images conditioned on text inputs. This output then gets used together with the text as the input to the second GAN in order to create a high-resolution image. The idea of stacking here is based on the notion that a second stage of conditioning forces the second GAN

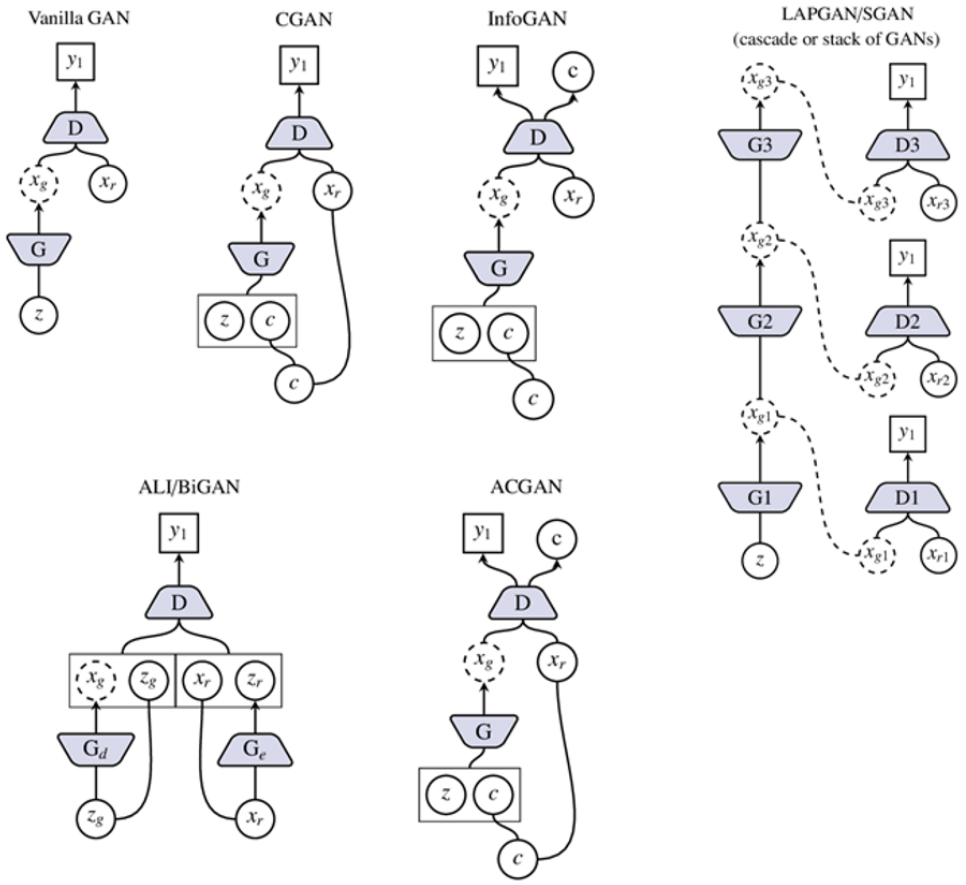


Figure 2.22: GAN Architecture Variants. Figure adapted from [127].

to learn text information that was omitted in the first stage and refers more to the details of the object.

Isola et al. [128] have proposed the pix2pix network, a cGAN with sparse regularisation used for image to image translation tasks. Rather than taking in a random noise vector z, the generator in pix2pix resembles a UNET which learns the mapping from one image to another. The objective function of the pix2pix network is as follows:

$$\mathcal{L}_{cGAN}(D,G) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x,z)))] \tag{2.11}$$

Furthermore, $L1$ distance is used:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[||y - G(x,z)||_1] \tag{2.12}$$

The final objective then becomes:

$$G^* = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \tag{2.13}$$

Isola et al. also make use of a PatchGAN discriminator which was initially introduced by [129] and works by selecting patches of the image which are classified by D rather than the image as a whole. The final output of D is formed by the average decision across all patches. pix2pixHD is an extension of the pix2pix network which uses a feature matching loss for high-resolution image synthesis and semantic manipulation.

Cycle-consistent adversarial networks (CycleGANs) [130] introduced a novel way of performing unpaired image-to-image translation. The idea is based on using two sets of generators and discriminators that translate images between two domains. Image X is mapped to image Y using function $G : X \rightarrow Y$ whilst image Y is mapped back to X with $F : Y \rightarrow X$. For both translations, discriminators D_X and D_Y classify samples into real or generated classes. The loss function is a combination of both adversarial losses as shown in traditional GANs as well as a cycle consistency loss which measures the similarity between image X and its mapping through Y and back to X. CycleGANs do not depend on paired images, making them highly applicable for a variety of unsupervised tasks including style transfer and object transfiguration. Conditional CycleGANs have been proposed by [131] which are cGANs with cyclic consistency.

Unsupervised Image-to-Image Translations (UNIT) networks proposed by [132] are another type of image to image translation network which work by putting two VAE-GANs [133] which are explained in section 2.5 back to back. Each VAE-GAN is hereby responsible for one image modality, though they share a common latent space.

RGANs have been used in a recurrent form for the prediction of next frames in videos and for producing real-valued multi-dimensional time series [134].

Least Squares GAN [106] aimed to solve the problem of gradient disappearance observed in traditional GANs as a result of using cross entropy loss. Instead a

least square loss function is minimized, which is equivalent to minimizing the X^2 divergence.

SRGAN [135] was the first implementation of GANs for super-resolution purposes for 4x upscaling factors. It is capable of creating high frequency details through a perceptual loss, a combination of adversarial and image content loss, given paired data.

Progressive GANs (PGGANs) [136] introduce a change to the GAN architecture whereby networks are progressively grown in depth during the training process. G and D start with a low number of layers which is increased step by step during training, learning increasingly finer details. Unlike greedy layer-wise training, however, the weights of the previous layers are not frozen during subsequent training. The training process is thereby kept stable and images of high quality can be produced.

StyleGAN's main idea focused on introducing latent noise at different levels during the generation process as opposed to just at the initial stage as seen in traditional GAN architectures. Additionally they utilize a mapping network to map latent space points to an intermediate latent space in order to influence the style of the output image. By tweaking the noise and latent spaces, authors claim to have control over high and low level details of the output. StyleGAN and particularly its followup paper Image2StyleGAN [137] reached a photo realistic image quality for human portraits.

Self-attention GANs (SAGANs) [138] aim to make use of recent developments in transformer architecture to introduce self-attention into image generation. The reason behind this step was to address the fact that cGANs struggled with the generation of some ImageNet classes more than others, particularly those with specific geometries like animals. This is likely linked to shortcomings of convolutional layers which only take into account local dependencies that are in the receptive field of the kernel. Increasing kernel sizes has long been proven to reduce computational efficiency. To overcome this issue, a self-attention layer was introduced in both G and D. Furthermore SAGANs utilise spectral normalisation for both G and

D. This is a form of regularisation in which weights within a layer are normalized to guarantee the Lipschitz requirement.

BigGAN [139] makes use of SAGAN's self-attention layers and uses them to generate large high resolution image outputs of up to 512 pixels which had not been done before.

Evolutionary GANs are networks where the adversarial training process is seen as mutation operations which evolve a population of generators from which the best performing generators are preserved [89].

Further application related developments of the aforementioned networks have been published in the field of classification [119], data generation, style transfer [140, 141], super-resolution [67, 142], reconstruction [143], photo editing [144] and text-to-image generation [145].

2.4.11 GAN Outlook

There are still several theoretical hurdles to be overcome in GAN research, for example the problem of global convergence which so far has not really been theoretically proven. The interpretability of NNs and GANs in particular leaves room for improvement, as well as the fine control of generator outputs. Lastly, there is a trade-off between computational cost and quality of GANs. BigGAN and StyleGAN have achieved photo realistic large scale examples, however, they had to be trained at considerable computational cost.

2.4.12 GANs in Medicine

Image synthesis

Annotated medical data sets are scarce and often difficult and costly to obtain. However, many deep learning networks require large training data sets to achieve competitive performance and data augmentation is not always sufficient to create enough variation within the data. Hence image synthesis is an important task in medical image analysis. It allows for the generation of large, realistic training data sets without the issues of anonymity or privacy concerns and can therefore be shared

freely. GANs are highly suitable for this task. Diller et al. [146] have used PG-GANs for generating synthetic short-axis CMR slices of patients with congenital heart disease. Random selections of the generated frames were subsequently manually segmented and used to train segmentation UNETs. The segmentation performance was compared to a UNET trained on real data. Whilst not being able to outperform the UNET trained on real data, the difference in segmentation quality was negligible. Further examples include [147] who uses an adversarial auto-encoder (AAE) to generate synthetic retinal vessel maps which are then fed into a cGAN to create full retinal images and [148] who use GANs to produce CT images of liver lesions for increased liver lesion classification.

Image Translation

Image translation is the conditional extension of image synthesis and GANs find widespread use for this task in the medical imaging field. There are several reasons for its clinical use, such as the fact that CT scans which are needed for radiotherapy treatments are more harmful to perform than MRIs, due to their associated radiation dose. Having realised that, [149] used paired CT and MRI volumes to learn a volume to volume translation process from MRI to CT. They implemented a fully convolutional network as the generator and added an image gradient difference term to the loss function to retain the sharpness of the CT.

Paired image data, however, are rare. Hence Wolterink et al. [150] used CycleGANs on unpaired CT and MRI images to enable MRI to CT translation. They found that this actually outperformed paired image sets. [151] combined cycleGAN image translation with a shape consistency loss through an added segmentation task. Chartsias et al. [152] use cycleGANs to translate cardiac CT into CMR data which is then used to successfully improve segmentation performance. CycleGANs have to be treated with care when used for medical image translation as shown in [153] if the target domain has an over or under representation of some classes, as this can lead to misdiagnosis.

General medical image translation, as done in [141], also incorporates non-adversarial losses, such as style-content loss to ensure perceptual similarity. Chartsias et al. [154] try to disentangle anatomical and imaging characteristic factors. They propose Spatial Decomposition Networks which factorise inputs into spatial anatomical factors and non-spatial modality factors. Through latent space arithmetic this allows them to do cross modality image translation, for example changing CMR to cardiac CT data. ScarGAN [155] uses unpaired images of healthy and scarred LV CMR scans to learn to transform healthy LV images into scarred ones.

Image Reconstruction

GANs have been widely used in image reconstruction. [156] used cGANs to recover missing CMR slices by mapping existing slices to a latent space with position vectors through a regression net. The latent vector corresponding to the missing slice is then projected to slice manifold, conditioned on intensity features through a generative net. Other methods use variations of the pix2pix structure or cycleGANs for a set of reconstruction tasks, such as denoising CT images [157], MR resolution improvement [158] or PET denoising [159]. It has also been applied directly to MR k-space reconstruction [160] and motion correction [161].

Image Segmentation

GANs are also used for image segmentation where the adversarial loss can also be seen as an adaptively learned similarity measure between the segmented outputs and the annotated ground truth [127]. So rather than measuring similarity in the pixel domain, the input gets projected to a lower dimensional manifold where similarity is computed which has a lot of parallels to perceptual loss. Applications of GANs for segmentation are found across modalities and organs. Rezaei et al. [162] use a cascade of cGANs for whole heart and great vessel segmentation whereby they also implemented an AutoContext Model for context-aware segmentation. SegAN [163] creates a multi-scale feature loss where features from several layers in the discriminator are combined. GANs have also been applied to segmentation

problems where unbalanced classes [164] or small data sets [165] are present. Further segmentation applications are found in [166] and [167].

Features from the generator and discriminator can also be extracted and used for image classification [168].

2.5 VAEs

2.5.1 Introduction

Variational Autoencoders (VAEs) [92] are a type of generative network that belong to the class of explicit density models, whereby the aim is to explicitly specify a target distribution through a defined set of parameters. They are also classed as a type of directed probabilistic graphical models (DPGMs). As previously discussed, many explicit density models tend to suffer from the problem of intractability when maximising the likelihood of the posterior. Mathematically this is equivalent to solving the following integral: $P(X) = \int P(X|z;\theta)P(z)dz$ [169]. VAEs circumvent this problem by approximating the posterior $p(z|x)$ through variational inference rather than maximising the likelihood directly. This is done by optimising the lower bound on the data log-likelihood $L(x;\theta) \leq \log p_{model}(x;\theta)$ [88]. VAEs make use of standard NNs which allows them to be trained solely with gradient-based methods without the need for Markov Chain Monte Carlo (MCMC) methods [170].

2.5.2 VAE Structure

Mathematically speaking VAEs differ quite significantly from standard Autoencoders (AEs). From a structural point of view, however, they show a great resemblance, hence the similarity in name. Akin to an AE, VAEs also consist of a parametric encoder and decoder, sometimes also known as inference and sampling networks, whereby both networks are NNs with optimisable weights phi and theta. The bottleneck between the encoder and decoder forms the latent space.

In contrast to AEs, where the latent space consists of single vector that learns a deterministic mapping between input and output, VAE latent spaces learn the parameters of a multivariate probability distribution, thereby making the encoder and decoder probabilistic rather than deterministic in nature, as seen in Figure 2.23. Often normal distributions are picked for the latent representation of a VAE and hence the learnt latent vectors represent the means and variances of the multivariate Gaussian. The latent space of a VAE is highly regularised and

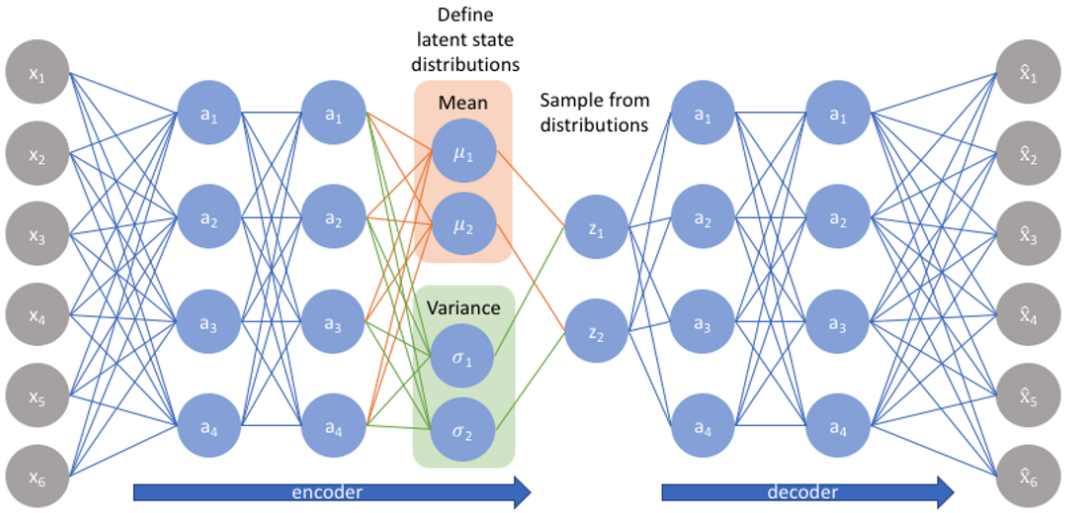


Figure 2.23: VAE Structure. Figure adapted from [171].

allows for effective sampling of points. These are passed through the decoder and mapped to data space to create new instances. Due to this property VAEs are considered generative networks.

2.5.3 VAE Loss

The goal of VAE training is for the estimated posterior $q(z|x)$ to approximate the real posterior $p(z|x)$ as closely as possible [172]. Also, from a generative network point of view, the marginal likelihood $P_\theta(x)$ is maximised. As this is intractable, the lower bound can be maximised w.r.t. ϕ and θ, the encoder and decoder weights respectively. This will allow for a good generative process. To measure and quantify the distance between the estimated and real posterior distributions, KL divergence is used. The goal is hence to minimize $D_{KL}(q_\phi(z|x)||p_\theta(z|x))$ with respect to the encoder weights ϕ. After some manipulation this turns to:

$$D_{KL}(q_\phi(z|x)||p_\theta(z|x)) = \log p_\theta(x) + D_{KL}(q_\phi(z|x)||p_\theta(z)) - \mathbb{E}_z \sim q_\phi(z|x) \log p_\theta(x|z) \tag{2.14}$$

Re-arranging the left and right hand side gives:

$$\log p_\theta(x) - D_{KL}(q_\phi(z|x)||p_\theta(z|x)) = \mathbb{E}_z \sim q_\phi(z|x) \log p_\theta(x|z) - D_{KL}(q_\phi(z|x)||p_\theta(z)) \tag{2.15}$$

Where the LHS is to be maximised which is done by maximising the log likelihood of generating real data given by $\log p(x)$ and minimizing the difference between the real and estimated posterior distributions. The loss can the re-written as:

$$\begin{aligned}\mathcal{L}_{VAE}(\theta,\phi) &= \log p_\theta(x) + D_{KL}(q_\phi(z|x)||p_\theta(z|x)) \\ &= \mathbb{E}_z \sim q_\phi(z|x) \log p_\theta(x|z) + D_{KL}(q_\phi(z|x)||p_\theta(z))\end{aligned} \tag{2.16}$$

This is known as variational lower bound which refers to the fact that KL divergence is always non-negative and thus $-\mathcal{L}_{VAE}$ is the lower bound of $p(x)$:

$$\mathcal{L}_{VAE} = \log p_\theta(x) D_{KL}(q_\phi(z|x)||p_\theta(z|x))_\theta(x) \tag{2.17}$$

Minimizing the loss means maximising the lower bound of the probability of producing real samples [173].

The $\log p(x)$ term refers to the reconstruction loss that is computed between the input data point and the reconstructed data point at the output of the decoder. Various losses, such as mean squared error (MSE) can be used for this depending on the application. For the estimated posterior $q(z|x)$ a variety of distributions can be chosen. The training data may not necessarily be following a distribution that is mathematically well-defined, therefore the encoder output is forced to approximate a known distribution [172], the multivariate Gaussian being a common choice. The prior over the latent variables $p(z)$ is usually set to be a centered isotropic multivariate Gaussian. Forcing the posterior to be close to a unit normal distribution is designed to organise the latent space to ensure continuity and completeness. Continuity means that two close points in latent space should map to two similar outputs in data space. Completeness means that for a chosen distribution a point sampled from the latent space should give meaningful content when decoded [174]. Keeping the covariance matrix close to identity means no punctual distributions occur, i.e. completeness is ensured. Keeping the mean close to 0 ensures continuity.

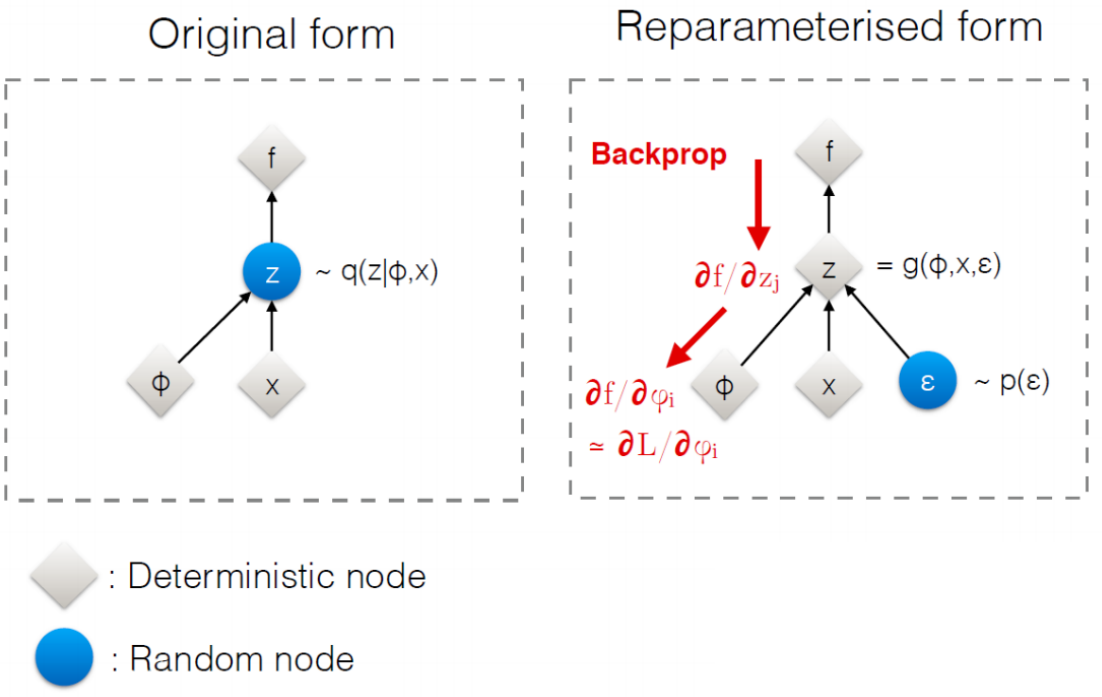


Figure 2.24: Reparameterization Trick in VAEs; turning a stochastic node into differentiable node. Figure adapted from [176].

2.5.4 VAE Training

During training a data sample x is fed into the encoder which maps it to a posterior distribution $q(z|x)$ over latent variables z [175]. The latent variables consist of two latent space vectors z_μ and z_σ which represent the means and covariances of the multivariate probability distribution $p_{model}(z)$. A point y is then sampled from this latent distribution ($p_{model}(z)$) and fed into the differentiable decoder $g(z)$. The decoder maps the latent space point back to the original data space. It models the conditional likelihood $p(x|z)$.

Backpropagation is a problem in this case though, as the process of sampling from the latent distribution is a stochastic step and therefore not differentiable. This means that gradient updates are unable to pass through the bottleneck.

2.5.5 VAE Reparameterization Trick

A reparameterization trick is used to overcome this problem. Instead of sampling from the specific latent space distribution, the sample is taken from a normal

distribution and augmented according to the formula $y = z_\sigma * x + z_\mu$ where x is the sample, z_σ is the latent standard deviation vector and z_μ is the latent mean vector. Hence:

$$z \sim q_\phi(z|x(i)) = N(z; \mu(i), \sigma(i)I) \tag{2.18}$$

$$z = \mu + \sigma \times \epsilon \tag{2.19}$$

where $\epsilon \sim N(0, I)$. Essentially the sample from the normal distribution gets scaled according to the latent parameters to produce what is effectively the equivalent to a direct sample from the latent distribution. This, however, allows for the stochastic part to be shifted onto the normal distribution sampling process and for backpropagation to be only applied to the learned parameters z_μ and z_σ.

2.5.6 VAE vs GAN

Training the encoder and decoder simultaneously forces the model to learn a predictable coordinate system that the encoder can capture [36] which makes it an excellent manifold algorithm. GANs have a clear disadvantage compared to VAEs: training them requires finding the Nash equilibrium of a game, which is a more difficult problem than optimizing an objective function. VAEs are also capable of obtaining a good likelihood, however, compared to GANs, they often produce blurrier images. It is argued that the problem of blurriness is closely related to maximum likelihood, i.e. the KL divergence in the loss function [36]. This is because the model gives a high probability not just to training data samples but equally to samples which may include blurry images. The reason for that is that a Gaussian is often chosen as the prior. This is effectively similar to having MSE as the loss in a normal AE; features of the input that occupy few pixels or that cause only a small change in brightness of the pixels it occupies are ignored, as argued by Theis et al. [177] and by Huszar [178]. Others argue that a VAE has difficulty capturing small-scale features such as textures and sharp edges due to the conditional independence of the output pixels, which significantly hurts both

the log-likelihood and the quality of generated samples compared to other models [175]. Further VAE issues include the tendency to only use a small subset of the dimension of the latent space [179]. The latent space representation does also not necessarily have an interpretable meaning.

At the time of the initial VAE paper, the main challenges to be addressed included the disentanglement of the latent space and the image quality during the generative process.

2.5.7 VAE Variants

Similar to GAN variants, a conditional VAE variant was developed [180]. This takes additional label information as input to allow for conditional output generation. During training the label is added to both the encoder and decoder input.

Adversarial Autoencoder (AAEs) [169] are very similar to VAEs, however, instead of using a KL divergence to regularize the latent space, AAEs use an adversarial loss. The latent space becomes an input to a discriminator network, as do samples from a multivariate unit Gaussian. The discriminator learns to distinguish them and the loss helps regularise the latent space. AAEs seem to be better at capturing the data manifold than standard VAEs, however, they do so implicitly, rather than explicitly. Furthermore no reparameterization trick is needed.

VAE-GANs [133] use a VAE structure, but with a GAN discriminator added after the VAE decoder to ensure that generated VAE samples are of high quality. The total loss function therefore comprises the VAE reconstruction and regularisation losses, as well as an adversarial GAN loss. Compared to a standard VAE, the VAE-GAN has shown to be better at overcoming blurriness [172].

VQ-VAE [181] also aims to achieve better reconstruction, however, of discrete latent variables. They implement a vector quantization algorithm with superior results to traditional VAEs.

InfoVAEs [182] aim to discover disentangled factors in latent space. In order to do so, they encourage large mutual information between the latent space and the image by adding a regulariser of maximum mean discrepancy (MMD). It can

lead to improved unsupervised representation learning based on the regularisation of the largest mean difference between distributions. Overall better performance metrics and stability compared to traditional VAEs has been observed [172].

β-VAEs [183] aim to disentangle the latent space and ensure that the variable in the latent space is only sensitive to one single generative factor and relatively invariant to other factors. This is achieved by adding the weighting factor β to balance reconstruction and regularisation losses in order to allow for better control of the output quality and orthogonality or disentanglement of the latent space. The loss function essentially becomes $\mathcal{L}(\theta, \phi, \beta) = \mathbb{E}q_\phi(z|x) \log p_\theta(x|z) - \beta D_{KL}(q_\phi(z|x)||p_\theta(z)$. This constraint limits the ability of latent information channels and emphasizes learning the statistically independent latent factors [172]. β-VAEs have been shown to perform much better at disentanglement than traditional VAEs.

Further disentanglement VAEs have been investigated by [184] which add a small amount of supervision during training.

Hyperspherical VAEs [185] are a recent VAE variant that does not use a Gaussian as the prior, as this can often lead to improper clustering in high dimensional data. Instead, the von Mises-Fisher distribution is used. This refers to a continuous probability distribution model on a circle, also called a circular normal distribution. Hyperspherical VAEs use a hyperspherical instead of a hyperplanar latent space which is better suited for capturing data with a hyperspherical latent structure, like protein structures. It has also been shown to outperform traditional VAEs in low dimensions. However, it may collapse in high dimensions [172].

Ladder VAEs are a VAE variant which was shown to be capable of learning a deeper and more distributed latent representation by combining the approximate likelihood and the data-driven prior latent distribution at each level of the generative model [186, 187].

2.6 CMR Analysis with Deep Learning

2.6.1 CMR Segmentation

Segmentation is often the first step in the CMR analysis pipeline. Manual segmentation is a long and tedious task which requires about 20 min per ventricle by a clinician [188] and comes with several difficulties. In the epicardium, poor contrast between myocardium and surrounding tissues, such as fat and the lungs, can cause segmentation difficulties. In the endocardium, gray level inhomogeneities of the blood pool and the presence of papillary muscles and trabeculations which have similar intensity levels to the myocardium make segmentation non-trivial.

Traditional Segmentation Methods

Prior to deep learning approaches, more traditional machine learning techniques were used to automate segmentation processes. These can be divided into methods using either weak/no or strong prior information. The former include image and pixel based methods such as thresholding, edge-based or region-based approaches. Deformable models, such as active contours and snakes could be used with either weak or strong priors. Atlas guided approaches, statistical shape models, Active Shape Models and Active Appearance Models required a strong prior.

Generally speaking in most methods a region of interest is extracted first, usually using object detection methods based on the fact that the heart is the only moving object in successive frames [189]. Hough transforms were commonly used to pick up the circular shape of the LV. When using thresholding, usually the endocardial contours were identified first [190], [191] and then used as a prior for the epicardial contour. Other methods such as radial lines [192] or gradient values [193] were also applied. Mathematical morphology was often used to address the difficulties caused by papillary muscles [193]. Benefits of no/weak prior methods was that there was no need for initial manual contours or any training of shape models or atlases. Generally though there was a trade-off between model performance and specificity.

Deformable models are a common approach for CMR segmentation. Snakes [194, 195], or Active Contour Models, worked by iteratively deforming curves

according to the minimization of an energy function which comprises a data-driven term, that provides information about object frontiers and a regularization term, that controls the smoothness of the curve. Variations of deformable models have been introduced over the years. Whilst they yield high segmentation accuracy, they require contour initialisation.

Strong prior methods include Active Shape and Appearance models which consist of a statistical shape model obtained by performing Principal Component Analysis (PCA) on aligned shapes [196]. Segmentation is done by overlaying the model on the image, and iteratively estimating rotation, translation and scaling parameters, whilst regularising the shape to remain within suitable limits for similar shapes. They have first been applied to cardiac segmentation in [197].

Lastly, atlas guided techniques have been used for CMR segmentation. They work by mapping the coordinates of an image to that of an atlas using a registration process [198] and then applying the obtained transformation to the atlas segmentation to obtain the final segmentation as seen in [199]. [200] use a multi-atlas-based segmentation approach with spatio-temporal registration for long-axis segmentation.

Deep Learning Segmentation Methods

Since the rise of deep learning in the 2010s, cardiac segmentation has been dominated more and more by CNN based approaches. This is largely due to improvements in methods and computational power, but also due to the availability of large annotated data sets like the ACDC set [201]. Whilst combined models using DL and deformable models [202] exist, the FCN and UNET approaches started to outperform traditional approaches. [203] first applied an FCN to cardiac short axis segmentation. Bai et al. [204] used an FCN to segment CMR data from the UK Biobank and derived a range of clinical parameters with high accuracy. Oktay et al. [205] incorporate prior knowledge about cardiac shape and location into their segmentation process for enhanced performance. Zhang et al. [206] develop the LU-NET which is a UNET version for ventricular segmentation whereby channels are re-weighted to give more important features more weight, similar to the SE-NET

model. Multi-scale inputs are combined with a UNET encoder to overcome the loss of pixel-location information. Furthermore to improve segmentation accuracy, the up-sampling stage is replaced by an un-sampling layer, whereby pixel location information from the encoder is used as guidance. Duan et al. [207] use landmarks in their bi-ventricular segmentation approach as a form of anatomical prior to overcome image artefacts, maintain spatial consistency and improve segmentation quality.

Whole heart segmentation is performed by [162] and [208] who combined faster R-CNN and UNETs for whole heart segmentation in cardiac CT volumes. [209] use a 3D UNET which applied multi-depth fusion to the original architecture to allow for context awareness. They also adapt the Dice loss with an additional focal loss to address class imbalance problems. [210] have addressed segmentation of myocardial infarction regions.

2.6.2 CMR Quantification

Following cardiac segmentation, cardiac quantification forms the next step. This involves determining a range of clinical parameters including LVEDV, LVESV, myocardial mass, stroke volume, EF, wall thickness and strain. This can be done by taking the result of an automated segmentation and directly calculating, but some work has proposed to estimated these directly using DL. [211–213] are examples of such an follow a similar approach. They first perform a spatial encoding step using CNNs to get a latent representation of the CMR frames. [211] then proposes a deep multitask relationship learning network for full LV quantification. Following the spatial encoding, RNNs are used to model temporal dynamics. The outputs from these are then used as inputs to both a Bayesian framework for estimating quantification parameters, as well as a classifier for detecting ED and ES phase. This is an important step for accurately determining EF and other clinical parameters. In a similar vein, [213] also use CNNs to first encode the spatial information of the cardiac sequence. They then, however, use an LSTM to decode the temporal information and regress to ED and ES phase information.

In [212] the encoding step is followed by a spatio-temporal CNN which regresses towards quantification parameters.

2.6.3 CMR Motion Analysis

Cardiac motion and functional analysis is another part of the CMR analysis pipeline and can be very useful in disease detection. Prior to deep learning regional LV myocardial function assessment was done via statistical MRI features whereby the similarity of distributions of features was used to characterise segmental contraction without the need for delineating LV boundaries [214]. Methods such as Linear Discriminant Analysis and support vector machines were commonly applied to do this [215]. Alternatively, functional myocardial abnormalities could be detected by analysing cardiac wall strains [216]. Furthermore atlasses of cardiac motion were compiled for the quantification of abnormal myocardial tissue velocities [217]. Myocardial wall motion was estimated using several traditional image processing approaches together with statistical shape models [218].

Deep learning has had a significant impact on motion analysis. [219] perform motion scoring which means identifying and classifying wall motion patterns into different groups. Their Cardiac-MOS network uses non-local CNNs [220] to classify myocardial wall movements into normal, hypokinetic, akinetic and dyskinetic. They claim that non-local relationships are important for finding subtle motion patterns and for identifying inter-class differences.

Qin et al [221] use a biomechanics-informed network for myocardial motion tracking in CMR. They argue that a registration process might result in deformations that are biomechanically infeasible. Hence they employ a biomechanical model to regularise registration results. Motion artefact correction is done by [222] using GANs.

Heart motion can also be used to better inform CMR segmentation. [223] use a joint learning system of both motion estimation and segmentation on 2D CMR Cine slices. The network uses two branches, a Siamese style recurrent spatial transformer network for motion estimation and a fully convolutional network for segmentation. By training them simultaneously the segmentation gets informed by the features

in the motion estimation branch and thereby achieves higher accuracy. [224] use a similar integration of heart motion into cardiac segmentation by augmenting a UNET structure. They integrated heart motion through the computation of the optical flow field of consecutive frames combined with a CNN module for LV localisation and self-attention.

Generative models for modelling cardiac motion have been proposed by Krebs et al. [225, 226], who have investigated probabilistic approaches of modelling cardiac deformation using generative methods including cVAEs. The generative models are used to learn low-dimensional probabilistic deformation models for cardiac image registration. [227] use a hybrid spatio-temporal network to generate dense displacement fields for temporal data to enable regional analysis of LV function. [228] combine image and patient metadata to enable the prediction of 3D cardiac shape parameters. They implement a statistical shape model to simplify shape complexity. The model passes image data of two cardiac views through FCNs and then concatenates them with metadata that has been passed through an MLP, similar to FiLM networks [229]. The output features are then passed through a CNN that regresses to a k-dimensional PCA space. This end-to-end model allows them to predict shape directly from a PCA space.

2.6.4 CMR Classification

Cardiac disease classification using deep learning has been attempted by Bello et al. [230] who use motion patterns to predict human survival. They start by segmenting cardiac frames to build 4D meshes which are subsequently passed through an autoencoder. From the latent space a survival prediction is made and a survival loss is added to the AE's reconstruction loss for training. Through the latent representation, modes of variation for varying degrees of survival prediction scores can be visualised in a form of latent space arithmetic. [231] try to classify myocardial hypertrophy from echocardiographic data.

Myocardial infarction classification was done in [232] and [233]. In [232] the authors introduced a framework for classifying and localizing myocardial infarction

areas by extracting features from fixed-sized patches of the LV. Feature extraction was based on intensity information and optical flow which described the static and dynamic characteristics of the regions respectively. They also used latent feature representation with a stacked denoising autoencoder (SDAE) to perform a pixel-wise binary classification for the identification of abnormal myocardial regions. [233] classifies regions of myocardial infarctions based on local and global motion. Local motion was captured using an LSTM to identify intensity changes between patches whereas global motion information was obtained from an optical flow field. Combinations of both showed to achieve the highest levels of accuracy.

[234] combined segmentation and classification of CMR slices. An ensemble of 2D and 3D UNETs was used for segmenting individual frames from which information was extracted that was then used in an MLP for disease classification. [235] use a form of semi-supervised learning of apparent flow to perform cardiac disease classification on cine MRIs. They generate a time-series of myocardial segments using a combination of apparent flow maps and cardiac segmentation. The pixel-wise apparent flow maps were employed to extract motion features and generated by computing the displacement between the ED frame and any other instant. The proposed method utilizes binary classifiers for CVD identification based on the extracted shape-related and motion-characteristic features.

Biffi et al. [236] proposed a method for learning interpretable anatomical features through deep generative models. They applied a 3D VAE framework to cardiac volume segmentations in order to learn a low-dimensional latent representation. From that they performed classification of several cardiac conditions. Additionally they implemented latent space navigation to explore anatomical variability learned by the latent space.

They continued their work in [186] by implementing ladder VAEs which, similar to UNETs, have skip connections between symmetrical levels on the encoding and decoding branch, however, in a ladder VAE these are parameterized. Their network allows for classification of clinical conditions at the deepest level of the VAE which therefore allows for direct visualisation of the classification in the latent

space. Moreover, the anatomical variability encoded in the latent space can be visualised. This hierarchical network allows for an interpretable shape analysis in the assessment of cardiac remodelling.

Related to this, Oktay et al. [205] show how the classification of dilated cardiomyopathy and HCM can be improved by using autoencoder-derived features from LV segmentations compared to just using PCA features.

2.7 Literature Review Conclusion

The field of deep learning really sprung to life in 2012 with the introduction of AlexNet and has progressed at a rapid pace since then. Deeper and more sophisticated architectures have been trained with larger data sets and more powerful GPUs, leading to ever increasing levels of performance. In many applications this resulted in networks starting to outperform human experts. Big breakthroughs in CNN developments, such as ZFNet, VGGNet and GoogLeNet were accompanied by time-sequence based methods such as RNNs and more recently attention methods and graph NNs. 3D networks flourished with the availability of larger data sets and GPU power and generative networks had their breakthrough following the initial GAN and VAE papers. After some teething problems, GANs and VAEs, including all their variants, began to establish themselves in the medical image analysis space. Here they were combined with UNETs to achieve even higher accuracies in image segmentation tasks, used in their conditional forms for cross-modality transfer tasks, such as CT to MRI transformations, or aided with reconstruction tasks such as missing slice filling or motion correction.

More specifically, shortly after work on chapter 3 in this thesis had been completed, it was elsewhere [146] shown that cardiac structure of patients can be captured from CMR data using GANs, hence validating the method in this chapter. More specifically, Diller et al [146] showed that by using PGGANs the cardiac structure of patients with congenital heart disease can be learnt from CMR scans. The performance of the network and the quality of the generated images were of a comparable level to the work in this chapter. Additionally, Diller et al manually segmented a set of generated images and used Dice scores to evaluate their performance. Furthermore, these segmentations were then used to successfully train segmentation models.

Furthermore, it was shown that VAEs can be used to learn interpretable anatomical features from cardiac volume segmentations. Latent spaces have also been used for exploring anatomical variability of cardiac shape and were used for disease classification. Yet a patient-driven sub-population based approach for

these methods is still lacking. In terms of cardiac function and motion it has been shown that probabilistic modelling using VAEs can be used to address registration problems between cardiac frames. Frame-to-frame prediction using GANs has been successfully presented, though not with a focus on cardiac ED to ES transformation. Motion patterns from the myocardial wall have been extracted from CMR data and classified corresponding to patient groups. Disease classification and clinical outcome predictions based on motion patterns have been presented, though not with the combinations of GANs and the context of myocardial infarction. For cardiac segmentation, particularly with regards to the large UK Biobank data set, Bai et al. [204] form the benchmark using FCN approaches. GAN based approaches have been used for different segmentation purposes, though not as extensively on UK Biobank CMR data.

In summary, this chapter has outlined the most relevant clinical and technical background information for the research in this thesis. Furthermore, it has reviewed the literature in the fields of deep learning and generative learning and reviewed applications of such methods in the cardiac image analysis space. A few limitations of the field and their relation to work in this thesis have been highlighted.

Looking ahead into the future, the directions in which the field will move are varied, though on the deep learning side common trends such as the increased use of graph-based networks and attention modules can be highlighted. On the cardiac side, DL based cardiac function models will be combined with biophysical and electromechanical models to incorporate pre-existing knowledge about anatomy and physiology into imaging data [221, 237] and to drive patient specific analysis [238]. Together these will enable more efficient modelling of CMR data and higher specificity in predictions which will subsequently allow for more nuanced and patient-specific cardiac function modelling.

3

Healthy Cardiac Structure - How to Capture It

Contents

3.1 Introduction 85
3.2 Data 86
3.2.1 Introduction 86
3.2.2 Pre-Processing Steps 86
3.3 GANs for Capturing Cardiac Structure 89
3.3.1 Introduction 89
3.3.2 GAN Structure 89
3.3.3 GAN Hyperparameter Optimisation 90
3.3.4 GAN Autoencoder Pre-training 104
3.3.5 "Optimum" GAN 106
3.4 VAEs for Capturing Cardiac Structure 112
3.4.1 Introduction 112
3.4.2 VAE Structure 112
3.4.3 VAE Hyperparameter Optimization 113
3.4.4 "Optimum" VAE 118
3.4.5 VAE Latent Space Analysis 120
3.4.6 New Data Generation 124
3.5 Conclusion 127

3.1 Introduction

The first step in understanding cardiac function is to understand cardiac structure, i.e. the typical appearance of cardiac anatomy in its static state. This can be achieved by learning the underlying data distribution of a CMR image data set. To do so, generative deep learning methods can be employed, as described in Chapter 2.

Generative networks are highly suitable for such tasks, as they are capable of capturing multivariate data distributions effectively. Generative adversarial networks (GANs) in particular are useful, due to the adversarial loss that they employ. This enables them to better reward the capture of realistic image features, such as high-frequency image detail like sharp contour boundaries, small papillary muscles or minor myocardial shape variations. The adversarial loss enables GANs to learn the discriminating features of a specific data set which make samples appear realistic, i.e. as if drawn from the true distribution. As a result, issues causing a lack of image realism, such as image blurriness, are hereby often reduced.

Whilst being powerful tools, GANs are notoriously difficult to train and the process is inherently unstable. Furthermore, the evaluation of GAN performance is non-trivial, due to the lack of ground truth data. Nonetheless, the suitability of GANs for learning cardiac structure from CMR images is promising and was therefore investigated in this chapter. Further generative methods such as Variational Autoencoders (VAEs) were also investigated and used as a comparison to GANs in this task. The relative advantage of VAEs centers around their easily accessible latent space which compresses data in a regularized and meaningful manner and can be used for visualising structural modes in the data set.

The overall aim of this chapter was therefore to investigate both network types in their ability to capture cardiac structure from CMR data sets.

3.2 Data

3.2.1 Introduction

The data used in this chapter stems from the Data Science Bowl Cardiac Challenge Data [239], henceforth referred to as "Kaggle Competition". This is a data science competition that was launched in 2015 with the aim of finding an automated method for accurately determining ejection fractions from CMR images. It forms a public CMR database that is comparatively large and extremely varied. The intention behind this competition was to encourage competitors to create robust solutions that are applicable across patient groups, scanners and acquisition protocols. The data set comprises patient data from across all age ranges, was sourced form numerous hospitals using various image acquisition modes and displays a large variance in image quality, anatomy and function. It also includes scans of patients with both normal and abnormal cardiac function.

The data set consists of 2D CMR scans from a total of 700 patients which have been split into training and validation groups of 500 and 200 patients respectively. For each patient several short-axis (SAX) scans are available. On top of that, most patient folders also contain long-axis (LAX) 2-chamber (2ch) and 4-chamber (4ch) scans. In the majority of cases, each CMR scan consists of 30 frames that cover the entire cardiac sequence, starting and ending at end-diastole (ED). In total, this adds up to more than 222,600 CMR frames with the data being stored in DICOM format. Images vary in their dimensions, with 256×192 pixels being the most common setup.

3.2.2 Pre-Processing Steps

Image Cropping

In order to create homogeneity between images and make them more suitable for subsequent use in DL networks some pre-processing steps were taken. Firstly, image dimensions of data samples had to be standardised. Therefore images were cropped to a 128×128 pixel dimension with the centre of the LV at the cropping centre. To locate the centre of the LV it was assumed that the difference between

adjacent frames in a sequence was due to motion and that motion occurs mostly in a symmetric way around the LV centre. Hence by computing the pixel-wise difference between adjacent frames along a sequence and taking the average of the centres of mass of each "difference" image, the average location of the centre of the LV could be determined. To further increase the robustness of the LV centre location, this method was performed for several mid-axial slices in the SAX stack from which results were averaged to determine an overall LV centre location for each patient.

The assumption that was made here is that the LV is the only moving component in the image. This is not entirely true, as each slice is taken during a separate breath hold, therefore perfect registration between slices cannot be expected. Furthermore, if a certain heart is beating asymmetrically, the location of the LV centre will be slightly shifted. Even though, by averaging the location of the centres of mass from all "difference" images, it can be assumed that a sufficiently accurate approximation of the LV centre can be attained. Furthermore, slight deviations in LV centre positions are not considered to be a problem, in fact they are desirable, as they aid with the generalisation of the data set.

Frame Selection

The large variance in patient backgrounds, data sources and acquisition protocols in the Kaggle Competition results in an inhomogeneous appearance of CMR scans across the data distribution. Whilst desirable for the actual competition, in this chapter it was decided to simplify further downstream data use by reducing data variability and homogenizing the appearance of the training set by only using mid-axial slices. This was achieved by choosing the median slice in the SAX stack of each patient, as well as one adjacent slice below and above that. Furthermore, only the ED frame, as well as the 3 adjacent frames either side in the sequence were chosen. Whilst this improved data set homogeneity, it did not achieve it entirely. Image samples taken from the data set after the previously mentioned filtering process had been applied can be seen in Figure 3.1. These clearly display the still existing variance in image appearance within the data set.

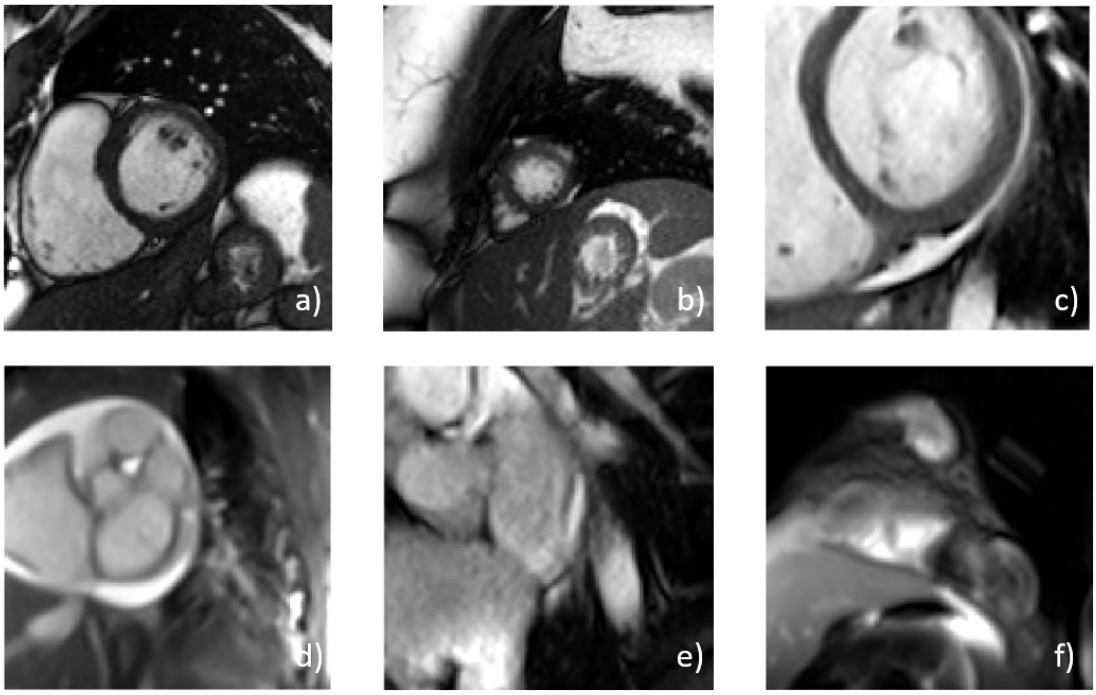


Figure 3.1: Example images from the Kaggle Competition data set. Even though the data set has been pre-processed to improve image homogeneity, a diversity in image quality and content can still be observed. A range of example images that capture the diversity are displayed here. Cardiac mid-axial SAX slices can range from "normal" scans which capture the ventricles in full (a), including some of the surrounding anatomy, to scans where the heart occupies either a very small (b) or very large (c) fraction of the image space. Additionally scans that fail to display easily identifiable cardiac anatomies are also included in the data set (d-f).

In summary, following post-processing and filtering steps a total of 8967 images were available for network training with a further 2744 images for validation and testing. No augmentation was performed on the images, as the total number of training and validation images was deemed sufficiently large. Image normalisation was performed to squash pixel intensities to a range of [0,1].

3.3 GANs for Capturing Cardiac Structure

3.3.1 Introduction

As outlined in the chapter introduction, both GANs and VAEs were investigated in their suitability for capturing cardiac structure. This section concerns itself with using GANs for that purpose. It begins by outlining the base structure of the GAN that was used which is an architecture derived from the DCGAN paper [119]. A GAN hyperparameter optimisation was then performed and evaluated. Following that, it was investigated whether pre-training the GAN generator could help boost GAN performance. Lastly an "optimum" GAN for CMR image generation was built and evaluated.

3.3.2 GAN Structure

The basic structure of a GAN, as described in section 2.4, consists of a generator network G and a discriminator network D which work against one another in an adversarial fashion to enable mutual learning. The generator hereby most commonly takes the form of a convolutional decoder, where a latent space vector z is transformed into an image or data instance by passing through multiple stages of convolutions and up-sampling steps. The base architecture of G can be seen in Figure 3.2. It closely resembles the architecture used in the original DCGAN paper [119]. A latent vector of length 1000 is initially passed through a fully connected Dense layer and reshaped into a square image. It then passes through several layers of convolutions followed by ReLU activations, batch normalisations and finally an up-sampling stage, before hitting a final convolutional layer with a single output channel which outputs the generated image.

The discriminator, as seen in Figure 3.2, commonly takes the structure of a convolutional encoder which transforms input images into binary classification outputs. It consists of 4 layers of convolutions each followed by dropout, leaky ReLU and batch normalisations layers, apart from the initial layer which does not have any batch normalisation in place. The convolutional layers have a stride of 2 and a

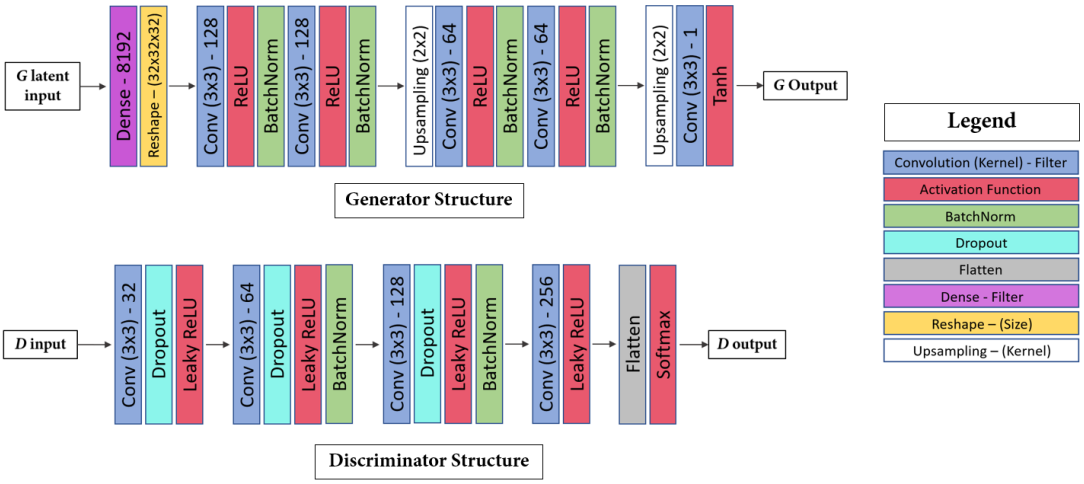


Figure 3.2: Structures for Generator G and Discriminator D networks.

kernel size of 3, thereby eliminating the need for max-pooling layers. The output of the last layer gets flattened and densely connected to a 2-node binary classification layer. The network employs a binary cross entropy loss in the discriminator. The learning rate is chosen as 10^{-4} and an Adam optimizer is employed.

3.3.3 GAN Hyperparameter Optimisation

Introduction to Hyperparameter Optimisation

Hyperparameter optimisation is an important step in the development process of deep learning networks and helps with achieving optimum network performance. If this step gets neglected, networks can fail to perform any meaningful learning. For example, networks can get stuck in local minima on the multi-dimensional error surface due to insufficient learning rates. Equally, if learning rates are too high or not well adjusted, networks can end up bouncing out of minima without ever converging to an optimum solution. Furthermore, badly chosen network architecture parameters, such as the number of layers, filters and convolutions can have a significant impact on how well the network is suited to the data set at hand; too little network capacity and the network will not fully learn the underlying data structure, too much network capacity and the network will quickly overfit and drastically reduce test performance. Each network and task has a specific

Table 3.1: GAN hyperparameters with associated value ranges and experiment number corresponding to results in Figure 3.4. Baseline values for each parameter are indicated in bold.

Parameters	**Value Ranges**	**Exp Num**
Number of Layers in G	[1, 2, **3**, 4, 5]	B
Convolutions per Layer in G	[1, **2**, 3]	D
Filter Scaling Factor in G	[1, 2, **3**]	C
Dimension of Input Latent Space to G	[10, 20, 50, 100, 200, 500, **1000**]	A
Number of Layers in D	[1, 2, 3, **4**, 5, 6]	E
Convolutions per Layer in D	[**1**, 2, 3]	G
Filter Scaling Factor in D	[1, **2**, 3]	F
Amount of Dropout in D	[**0**, 0.1, 0.25, 0.5]	H
Learning Rate	[1e-2, 5e-3, 1e-3, 5e-4, **1e-4**, 5e-5, 1e-5, 5e-6, 1e-6]	J
Batch Size	[16, **32**, 64, 128, 256]	I

combination of hyperparameters that will result in optimal network performance. These parameters are, however, often not very intuitive due to the "black box" nature of most neural networks and hence require careful investigation. For GANs, this problem is even more substantial, as GANs are inherently unstable during training. The choice of hyperparameters likely makes a more significant impact, as GANs are playing an optimization game rather than converging on an objective function.

Optimisation Method

To find the optimum set of network hyperparameters a full grid search of all possible combinations would theoretically ensure success. However, as this is a multivariate problem, extensive grid searches are computationally not feasible. Instead, a robustness test and single parameter investigations were performed. The previously outlined setup which is based on the well performing DCGAN was used as a base line setup. From here, individual parameters were changed one at a time through a range of suitable values and network performance was analysed. This was assumed to not only inform about the robustness of this setup, but also to potentially give an insight into the effect of individual parameters on network performance and allow for the identification of trends. To assess the performance of each hyperparameter setting,

the performance metrics which are discussed in the next section were applied. The investigated hyperparameters and their associated value ranges are given in table 3.1. Baseline values for each parameter are shown in bold. Alternative strategies for hyperparameter optimisation include methods such as Bayesian Optimization. This uses Bayes Theorem to direct the search towards finding a minimum or maximum of an objective function. Previously mentioned methods such as genetic searches or random searches could also be employed. These strategies could offer better approximations towards an optimum than the implemented single parameter line search, however, given time constraints this was not investigated further.

GAN Performance Evaluation - Qualitative

Due to the lack of ground truth data, straightforward performance evaluation of GANs was not possible. However, several other methods of GAN performance evaluation can be found in the literature [107]. Though many are ill-suited for the work in this chapter, a selection of several suitable qualitative and quantitative methods for performance evaluation was employed.

Qualitative evaluation is often seen as a gold standard and an easy to implement solution to the problem of GAN evaluation [107]. Methods include, but are not limited to: Rapid scene categorization, evaluating mode drop and mode collapse, nearest neighbour finding, rating and preference judgement [107]. In this chapter a combination of rapid scene categorization and rating was employed, due to its straightforward implementation and the availability of qualified assessors. Hereby samples of generated images from each GAN hyperparameter experiment were evaluated by a group of experts who judged the realness of generated images in two ways.

Firstly, a mixed collection of real and generated samples was given to experts for rapid scene classification. This required the experts to classify each image individually as real or fake in rapid fashion, i.e. without having the time to examine the images in detail. Accuracy scores from the test sets were averaged across all experts to produce average scores for each GAN experiment. A score between 0-1 could be achieved with 0.5 forming the optimum as it indicates that 50% of the

time real and fake images were incorrectly classified, i.e. the assessor is perfectly confused by the generated images. A score of 1 indicates no confusion at all, i.e. the generated samples are unable to trick the assessors into thinking they are real.

Secondly, experts were given sets of samples of generated data which they were to rate on a scale of 1-5 with regards to their realness. A score of 1 hereby stands for a very unrealistic sample, such as random noise or unidentifiable blotches, whereas a score of 5 stands for a sample that is indistinguishable from real samples. Scores in between the extremes scale in meaning accordingly. The experts were given a set of real images as a baseline for comparison. Image characteristics such as blurriness, contrast, gray level intensities, realistic anatomical geometry, lack of identifiable features and image artifacts should be considered during the rating process. Examples for each rating can be seen in Figure 3.3. A score of 2 for example would be given to images that are no longer just noise, but display contours and structural elements, though these are not necessarily identifiable as cardiac structures. A score of 3 is given to images which begin to show cardiac structures, such as LVs and RVs, however, do so whilst also being either very blurry, anatomically very inaccurate or display other artefacts and noisy features in the images. A score of 4 is given to images that display cardiac structures with a high level of sharpness, correct contrast and realistic gray level distribution. These images are, however, still identifiable as fakes, due to occasional blurry image patches or features of anatomical incorrectness, such as broken LV myocardiums. A score of 5 would be given to images that are largely indistinguishable from real images. Scores are averaged across all shown samples and across all experts to generate an average score for each GAN experiment.

The images that were used for the qualitative rapid classification assessment were picked as follows: For each hyperparameter value experiment, the three epochs that were judged to produce the best images were selected. From each of those epochs, three images were generated with random noise inputs. This means a total of 9 fake images were selected per hyperparameter value experiment. Altogether this summed

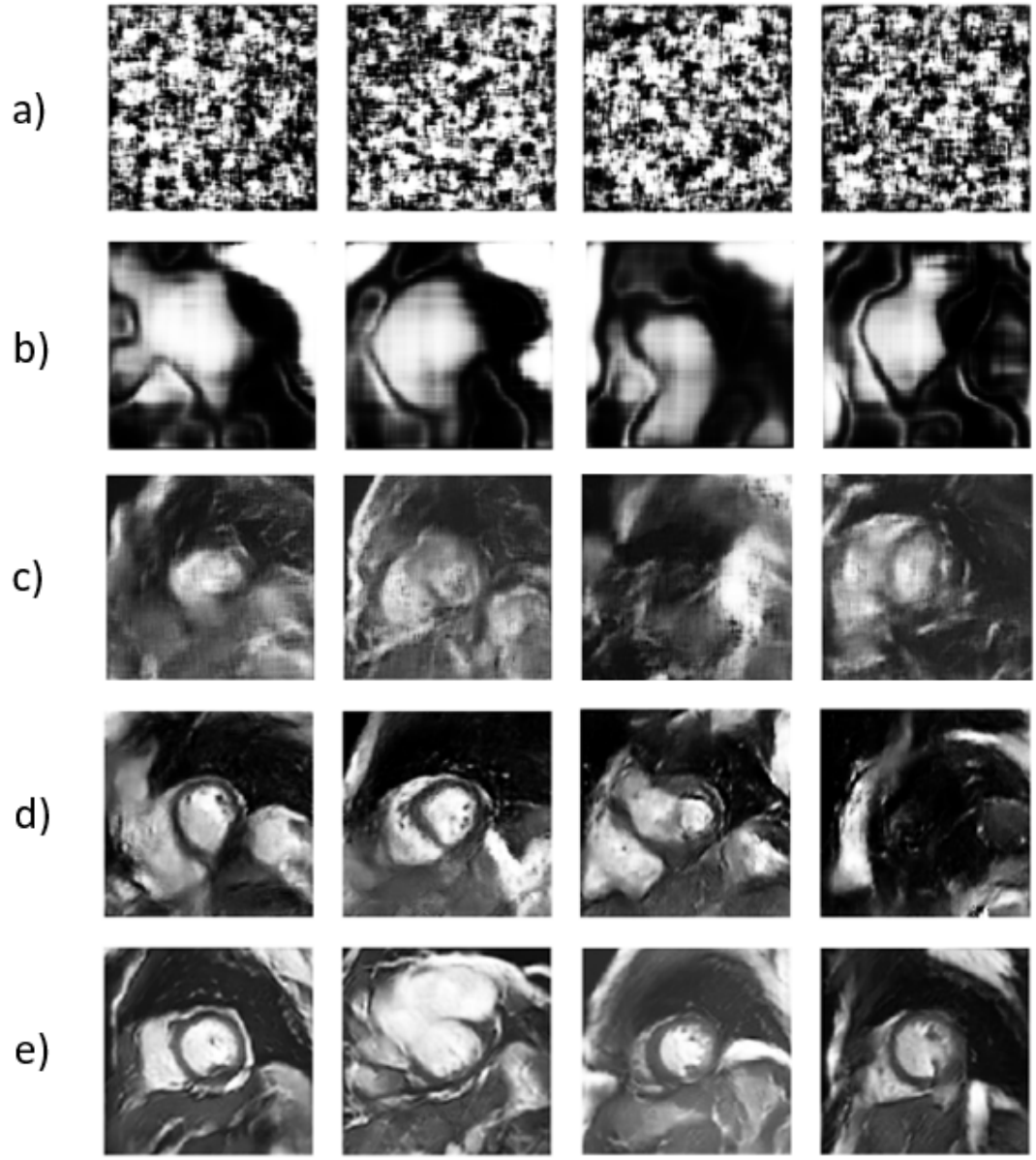


Figure 3.3: Example images from the qualitative rating assessment with scores of 1-5 in rows a)-e) respectively showing characteristic image features for each score.

to a total of 432 fake images for classification. An equal number of real images were picked at random from the real validation set and the two sets were mixed at random.

A group of 3 researchers who work in MR analysis, including the author, acted as assessors. Classification and rating scores were recorded separately for the author and the other experts in order to investigate for inter-operator variability. For the rapid scene categorization method images were displayed individually on a screen during the assessment, at which point assessors had to press one of two keys on the keyboard to indicate their classification. A set of additional real images for reference was not shown during this assessment.

For the qualitative rating assessment, a 4×4 grid of generated fake images was shown to the assessors, at which point they had to give a rating of 1-5. No time limit was applied for making the decision. The reason for using a grid, rather than a single image, was to smoothen out the rating scores across the shown distribution. This was not done for the rapid scene classification part, as a single "bad" image in a grid would give away the nature of the image set too easily. For each hyperparameter value experiment, 3 grids of images were shown to the assessors. Image grids from all experiments were randomised in their order to appearance to prevent biases.

GAN Performance Evaluation - Quantitative

For quantitative evaluation, the Inception Score is one of the most widely used metrics [107]. Using the Inception Score as a metric in this task, however, was inappropriate as it is solely designed for the use on natural image generations and is not easily transferable to other image domains. It works by analysing the confidence of classification of a generated image into one of the imageNet classes after having been fed to the last classification layers of the InceptionV3 Net. The more confident the network is in its classification into a single particular class, the higher the Inception Score, as it has assumed that the image is a realistic example of that specific class given that it is sharing a lot of its high level features. While a generated medical image might also share a lot of high-level features with an

ImageNet class, the Inception Score becomes meaningless as it does not signify anything about the realisticness of said image within its own domain.

The Frechet Inception Distance [108] (FID), as introduced in chapter 2, was more applicable in this case as it is not so much the similarity between medical and natural images that matters, but the similarity between generated and real image populations within the same domain. Even if the features used in the InceptionV3 are not directly suited for the medical domain at hand, they are equally ill-suited for the real and generated populations, introducing, if anything, a systematic error rather than a random one. Given that the aim was to compare different networks, absolute scores are irrelevant, only relative scores matter. Therefore it was believed that the FID would give some indication of the performance of the GAN.

A second, more task related, quantitative metric was employed as an additional tool for judging GAN performance. This was based on applying a pre-trained segmentation network to the generated cardiac frames. From the segmentation output of said network, myocardial masses and LV volumes, represented by the number of myocardium and cavity pixel labels in the segmentation, were recorded. The distribution of those values was then compared to the distribution of real images. This was done using the Kolmogorov-Smirnov (KS) test from which p-values for the similarity of the distributions could be calculated and used as a score. The KS-test is a non-parametric test that can be employed to test the similarity of two distributions. It effectively tests the probability that two sets of samples were drawn from the same probability distribution. It does so by quantifying the distance between their respective cumulative distribution functions. From this statistic a p-value can then be calculated. A high degree of similarity between distributions would indicate strong similarity between real and fake data sets and hence speak for the quality of the GAN's performance. To account for varying sizes of myocardium and cavity due to the slice location within the heart and to normalise the data, the ratio between myocardium mass and cavity volume was used instead.

There are, however, several shortcomings with this method. A bad segmentation could still lead to a non-zero amount of pixels being incorrectly labelled as my-

ocardium or cavity and matching a real image in terms of their relative distribution, thereby producing a misleading result. This can, to a certain extent, be limited by comparing the centre of masses of the segmentations which in the case of a good segmentation on a real SAX image should lie closely together. Furthermore, whilst the KS-test is a commonly used metric, due to its sensitivity to the differences in both location and shape of cumulative distribution functions, it also has its weaknesses. When the number of samples N gets large, the calculated p-value is almost always below the significance level, hence rejecting the null hypothesis that samples were drawn from the same distribution and thus indicating that distributions are different, even if they are only ever so slightly dissimilar. This is a problem as it reduces the meaningfulness of the p-value, making the the KS-test less reliable as a metric.

Due to computational and time constraints, only FID and qualitative metrics were applied for the hyperparameter optimisation experiments. The segmentation method was only applied to the evaluation of a final optimised model, as described in section 3.3.5.

GAN Hyperparameter Optimisation Experiments

All hyperparameters described in table 3.1 were investigated for their effect on GAN performance. The network performance for each hyperparameter setting was measured by both the FID and the qualitative methods, as described in the previous section and was bench-marked against the performance of the basic network setup.

A separate experiment was run for each value within the hyperparameter ranges in order to isolate the effect of each parameter on network performance. Between experiments, only a single parameter was changed from the baseline setup, with all other parameters taking the values of the baseline setup. For each experiment, the GAN was trained on the full training data set and the validation set was used for calculating the FID scores, as well as serving as the qualitative benchmark for image quality rating and classification.

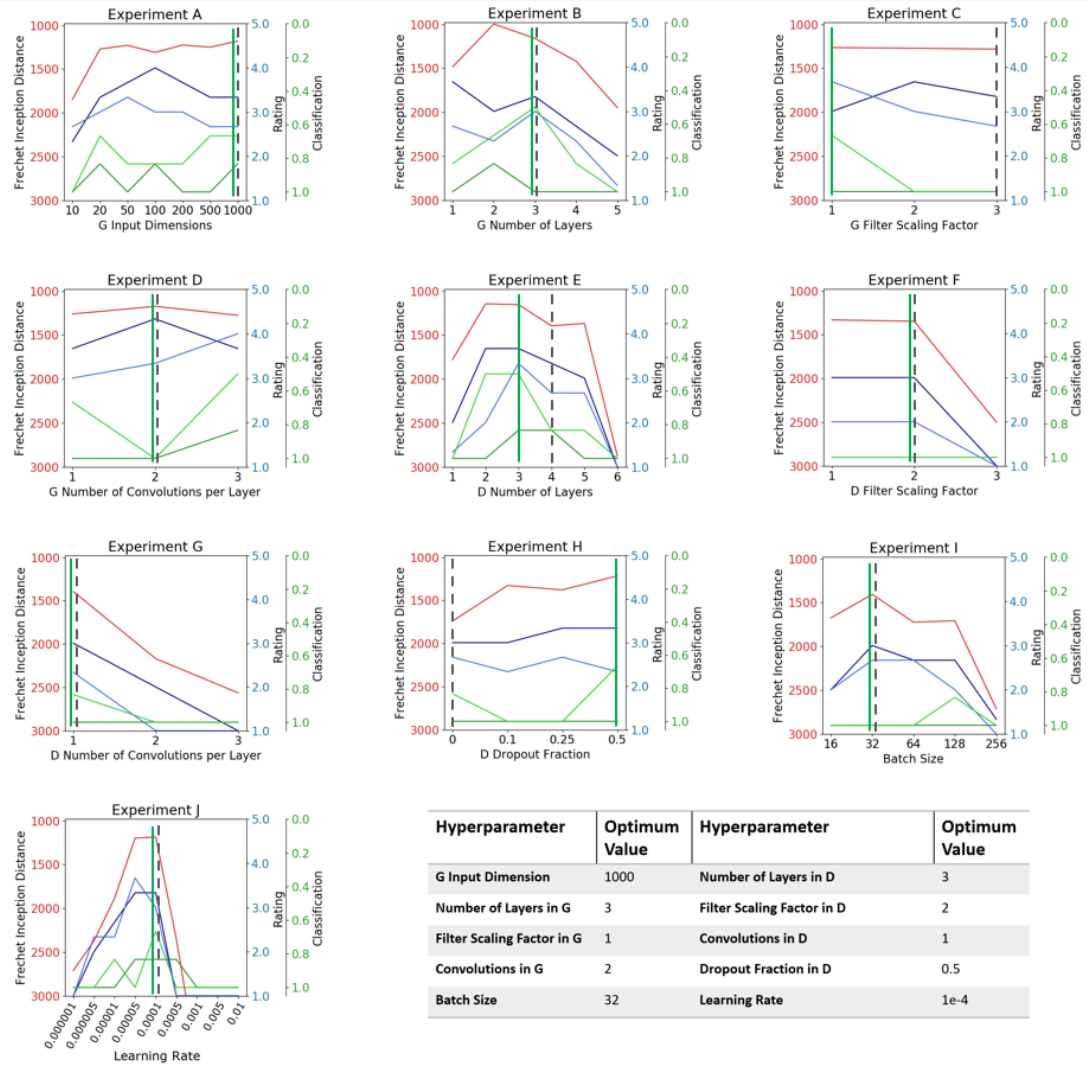
Since GAN loss curves are often not very indicative of convergence, a strategy of using early stopping triggers at a point of overfitting was not suitable for this task.

For networks with objective functions this point usually occurs when validation performance begins to drop despite improving training performance. Instead, for each hyperparameter experiment the GAN was trained for a fixed number of epochs, in this case 300, which formed a compromise between maximum performance and computational and time resources. Throughout training of each hyperparameter experiment the validation performance was tested at regular intervals and the top 3 validation scores were recorded. The average of these was then taken as the final result of that particular hyperparameter experiment to reduce the effect of one-off performance outliers.

GAN Hyperparameter Optimisation Results

Results of the GAN hyperparameter optimisation can be found in Figure 3.4. Experiments A-J hereby covered the hyperparameters as listed in table 3.1. On each graph red lines indicate minimum FID scores for the individual parameter values, dark blue and dark green lines represent the rating and classification scores respectively, as judged by the author, whilst light blue and light green lines represent the rating and classification scores respectively, as judged by the group of experts. Lower FID scores indicate better performance, as do lower qualitative classification scores. This is because lower classification scores imply that the assessing person has been more frequently tricked into assigning a real label to a fake image. The classification score has an optimum at 0.5 which indicates that 50% of the time real and fake images were misclassified, i.e. the assessor is perfectly confused. For qualitative ratings, a higher value indicates better performance. The vertical dashed grey lines indicate the base line value for each parameter prior to optimisation, whilst the vertical solid green lines indicate the chosen optimum value. Optimum values are also summarised in the attached table.

Additionally, Figure 3.5 displays example images taken from across all hyperparameter experiments to visualise the range of image quality produced and their corresponding FID scores for better understanding. An overall minimum FID score of 995.5 was achieved across all hyperparameter experiments.



Hyperparameter	Optimum Value	Hyperparameter	Optimum Value
G Input Dimension	1000	Number of Layers in D	3
Number of Layers in G	3	Filter Scaling Factor in D	2
Filter Scaling Factor in G	1	Convolutions in D	1
Convolutions in G	2	Dropout Fraction in D	0.5
Batch Size	32	Learning Rate	1e-4

Figure 3.4: Results of GAN hyperparameter experiments A-J. The red line indicates the FID score. The dark blue and dark green line represent the rating and classification scores respectively, as judged by the author, whilst the light blue and light green lines represent the rating and classification scores respectively, as judged by a group of experts. Note that all axes have been chosen so that better scores are at the top of each graph. The vertical dashed grey line indicates the base line value for each parameter prior to optimisation, whilst the vertical solid green line indicates the chosen optimum value. Optimum values are also summarised in the attached table.

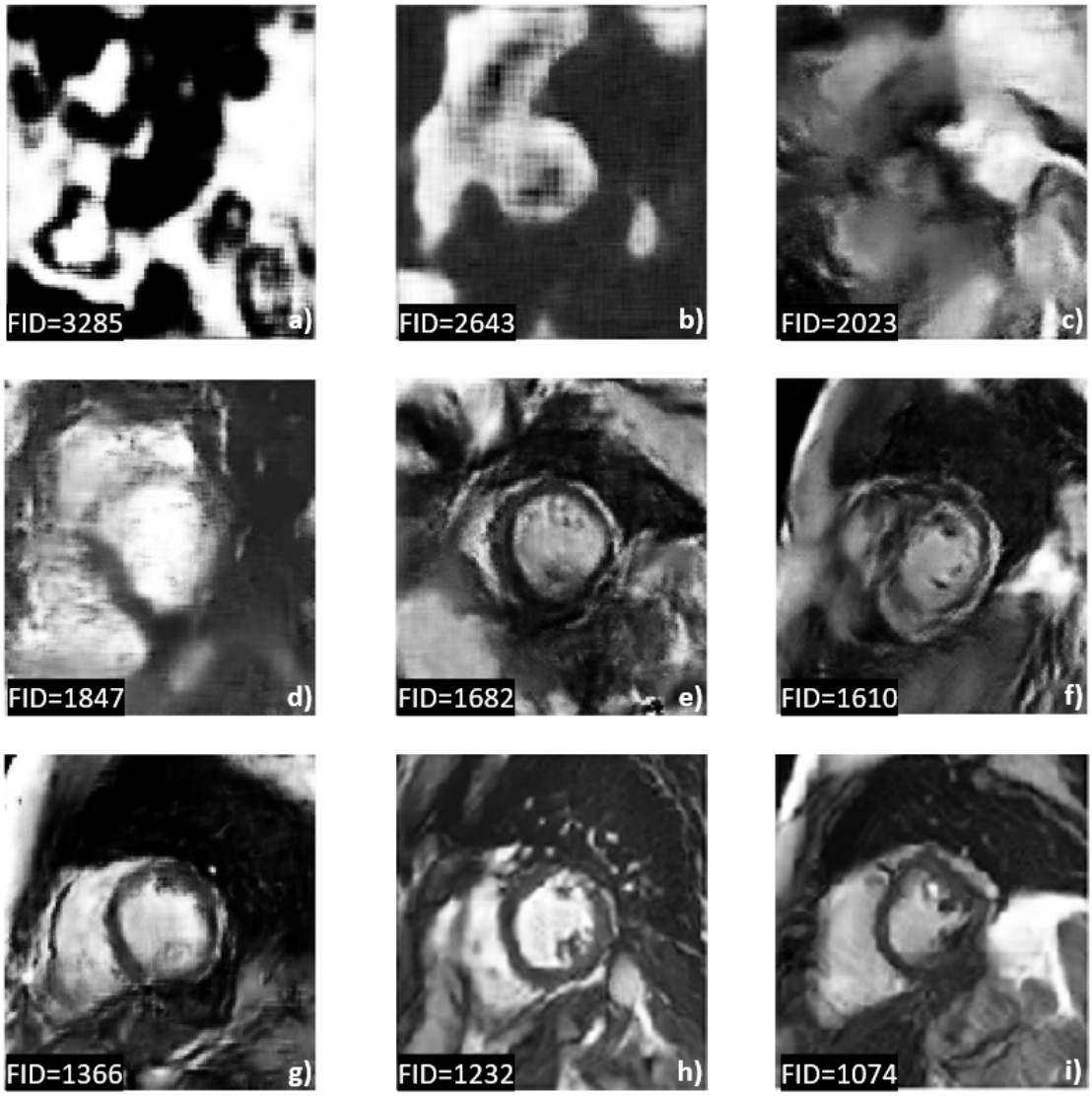


Figure 3.5: Example images from hyperparameter experiments, including corresponding FID scores. Images have been chosen from different experiments to highlight the correlation between FID score and image quality & content.

GAN Hyperparameter Optimisation Discussion

Generally speaking, based on results in Figure 3.4, there is decent agreement between metrics across all hyperparameter experiments. FID scores, however, proved to not always be perfectly correlated with image quality. When choosing the best images from each experiment to perform qualitative rating on, it was found that the images with the lowest FID scores, which were initially assumed to be of the highest qualitative standard, were in fact not necessarily so. Low FID score images

seem to contain a comparatively larger degree of high-frequency image features. Whilst this is usually a strength of GANs with respect to other methods such as VAEs, in the context of CMR data, it can have a negative impact on the realness of image appearance. This is due to the fact that CMR scans often contain a lot of smooth image regions corresponding to the internal structures of organs or bones. Therefore, to account for this phenomenon in the image selection process, if images with a higher, i.e. worse, FID score were found to sufficiently outperform those with the lowest FID score in terms of their qualitative appearance, the former were chosen for the final qualitative assessment instead. Despite this limitation FID scores show a close correlation with qualitative scoring, meaning that in most occasions images were chosen from the epoch with the lowest FID score.

During GAN training, the output largely followed a similar development pattern in its appearance as seen in images a)-i) in Figure 3.5. Initially starting with noisy images, rough shapes or blobs of largely black or white colouration would begin to appear which would then slowly begin to resemble cardiac shapes with rough anatomy. At this stage, smooth gray level variations would dominate the image. While they gave realistic looking contrast and clear edge and region definition, the anatomical accuracy lacked realism at this stage. Further on in training, images would usually gain more speckles and high frequency features. The anatomical accuracy would improve, however, the amount of small contrast variations within organs would take away from its realistic appearance. This often turned out to be a significant trade-off when it came to qualitative judging.

From the qualitative rating scores, it can be observed that the author's rating scores consistently outperformed those of the experts, indicating some degree of personal bias. This is not so much of an issue, as the absolute scores of the rating are irrelevant, as long as rating consistency across experiments is maintained. The author's classification scores are consistently higher than that of experts. This was to be expected, as having experience of working with GANs and CMR data sets, as well as understanding the failure modes of the networks helps with identifying irregularities more quickly which in turn makes classification easier.

Based on the analysis of the individual experiments, experiment A would indicate that a generator input dimension of less than 20 yields poorer results than larger dimensions. This is reasonable when considering the manifold hypothesis [240]. This states that real world high dimensional data can be well represented in low dimensional space, however, as this space becomes smaller, the ability to represent high-dimensional data disappears. This is likely the reason why network performance seems to decrease towards the lower end of the parameter range. There is, however, little agreement between the different metrics for larger dimensions that might indicate a trend towards an optimum value. Qualitative metrics slightly favour mid range values, while a lower FID score is given for upper range values. Computationally speaking lower dimensions are significantly more favourable, as the input layer of the generator is densely connected to the second layer. These fully connected layers are computationally very expensive, so a reduction in input dimension is useful. Alternatively using an input dimension that has a natural square root would allow for the dense connection to be eliminated and the input could be directly reshaped into a 2D grid. However, this change of architecture could have other effects on network performance which have not been investigated here. An input dimension of 1000 was kept as the optimum.

Results from experiment B would indicate that an excess of 3 layers of network depth for the generator leads to sub-optimal results. Whilst quantitative and qualitative metrics disagree on the likely value for optimum performance, with FID scores suggesting 2, compared to the quantitative metrics' choice of 3 layers, they do agree on the decreasing performance for networks exceeding 3 layers. Network depth is related to network capacity which, if too large, can lead to vanishing gradients. Of note is that for a value of 3 layers in the generator, quantitative classification indicates a score of 0.5 which represents maximum confusion of the assessor. Three layers were kept as the optimum.

Experiment C would indicate that the number of filters per layer in the generator does not have a strong effect on network performance. Metrics roughly align, with a slight preference for fewer filters. The same argument relating to network capacity

as for experiment B can be made here, hence a scaling factor of 1 was chosen as the optimum. Experiment D does not seem to display a strong correlation between network performance and the number of convolutions per layer in the generator. Interestingly, metrics seem to exactly disagree. Whilst qualitative classification scores lowest of 2 convolutions, FID and rating seem to indicate better performance. Overall, this does not provide convincing evidence for overall trends for this parameter, hence the baseline value was kept.

The results from experiment E seem to indicate a strong agreement between metrics and show that network performance deteriorates for very low and very high numbers of layers in the discriminator. Values between 2 and 5 layers are largely similar with a slight preference for 2-3 layers, hence 3 layers were chosen as the optimum. Experiments F and G both show good agreement between metrics and indicate a significant drop in performance for higher values of discriminator filter numbers and convolutions per layer. Together with the results from experiment C and D, this makes for an interesting result. In most discriminative deep learning networks, such as CNNs, an increase in network capacity, given sufficient data, usually results in improved network performance. However, the GAN seems to not adhere to this trend. Base line values were kept for these parameters.

Results from experiment H seem to indicate slightly better network performance for high dropout levels, although the variation across the value range, as well as the agreement between metrics is low. Interestingly zero dropout seems to perform quite badly on a quantitative level. A value of 0.5 was used as the optimum for dropout. Experiment I has good agreement between metrics and indicates low network performance for batch sizes above 128, with an optimum at 32. Performance also seems to drop slightly for low batch sizes of 16. This agrees with the theory of gradient descent. A compromise between stochastic gradient descent which updates the weights after each sample and batch gradient descent which takes the average update across all samples needs to be made to ensure efficient movement towards the global minimum without being slow or bouncing around too much.

Lastly, results from experiment J display a strong dependence of network performance on the learning rate. This is to be expected. GANs are not optimisation problems, but games, they therefore do not converge to a global minimum but a Nash equilibrium or saddle point. While training regular CNNs will yield decent results even for low learning rates, given that the training goes on for long enough, GANs are more sensitive. The learning rate of 10^{-4} was chosen as the optimum. A summary of all optimum parameters is given in the table in Figure 3.4.

3.3.4 GAN Autoencoder Pre-training

Methods

In addition to the hyperparameter investigation, the use of a pre-trained autoencoder (AE) as the GAN generator was tested. It was hypothesised that this could improve GAN performance. An AE learns to nonlinearly compress and map CMR input images to a latent space from which it then reconstructs them. Once trained, the decoder part of the AE could be used as a pre-trained generator for the GAN. The idea hereby is that this step might reduce the risk of vanishing gradients during GAN training. This is a common problem in the early stages of GAN training and is often caused by the discriminator being much better at distinguishing real from fake data than the generator is at creating realistic fake data. This causes generator gradients to vanish and training to stall. By using a pre-trained autoencoder, the hypothesis was that this could boost initial generator performance into producing more realistic images and thereby create more stable GAN training, as well as overall better and faster performance. The architecture of the AE decoder matches the architecture of the GAN generator to facilitate a seamless replacement of generators in the GAN architecture.

Autoencoder Pre-training Experiments

To improve GAN performance, an AE was pre-trained, from which the decoder was then used as the GAN generator. The AE decoder matched the baseline architecture of the GAN's generator to make easy network substitution feasible.

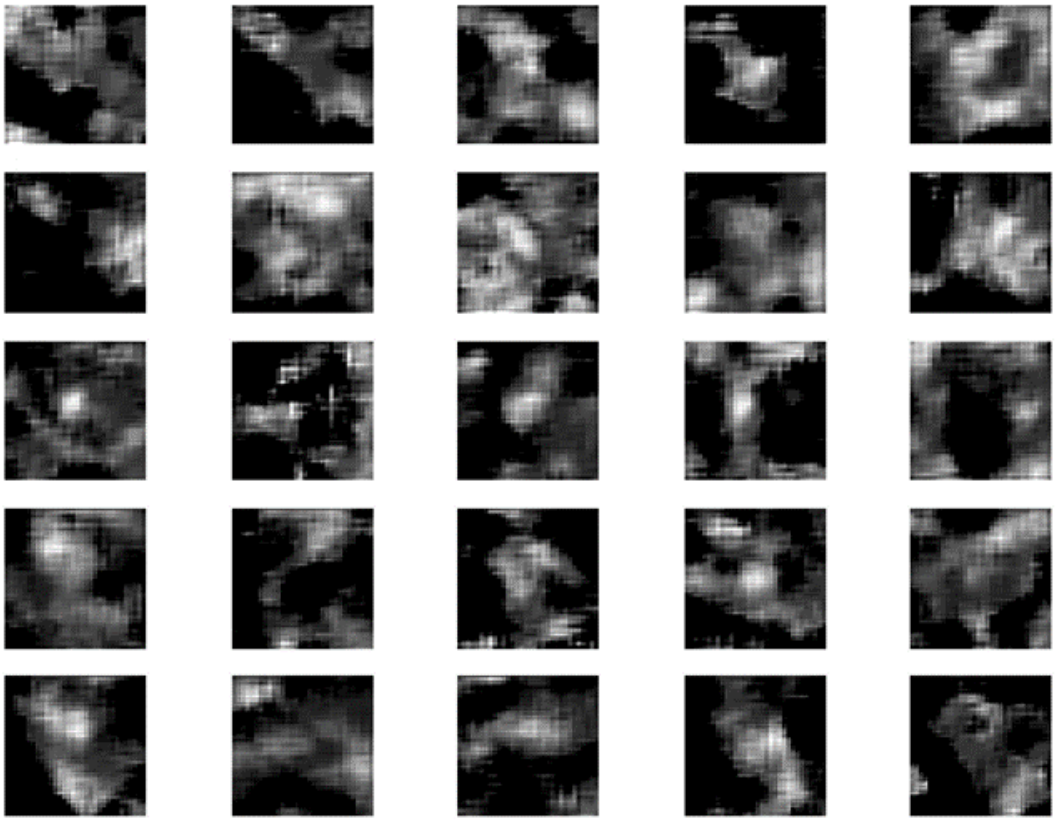


Figure 3.6: Example image results from using an AE pre-trained generator in the GAN.

The decoder forms a symmetrical mirror image to the encoder part of the AE, only with upsampling layers rather than maxpooling layers. The latent space consisted of a vector of 128 nodes and was fully connected to the flattened vectors of the encoder output and decoder input. All other hyperparameter matched the baseline parameters used for the basic GAN. Adam optimisation and an MSE loss were used, with a learning rate of 10^{-4}. The training data set was used for training, and the training was stopped when overfitting started to occur.

Autoencoder Pre-training Results

The Autoencoder pre-trained GAN achieved a minimum FID score of 3089 (for reference of FID scores and associated image quality use Figure 3.5). Qualitative rating and classification scores of 2.2 (out of a maximum of 5) and 1.0 (i.e. 100% with 50% being maximal confusion of the assessor) respectively were given by the

author, whilst scores of 2.1 and 0.95 were given by the group of experts. Example image outputs from the network can be found in Figure 3.6.

Autoencoder Pre-training Discussion

As demonstrated by the low quality of images in Figure 3.6 and the corresponding network performance results, it can be argued that using a pre-trained Autoencoder leads to suboptimal results compared to when the same network set-up is trained from scratch. An AE is trained end-to-end without any added regularisation of the latent space. In contrast, a VAE, as explained in more detail in section 3.4, uses a probabilistic mapping with an additional regularisation loss that aims to enforce a Gaussian distribution of the latent space. This is done to ensure continuity and completeness of the latent space, meaning that when using samples from a normal distribution as latent space inputs meaningful output tend to be generated. Since the AE does not have such regularisation, there is a chance that continuity and completeness of the latent space are insufficient, leading to unrealistic generated samples.

3.3.5 "Optimum" GAN

"Optimum" GAN Method

The hyperparameter optimisation process did not yield a global optimum parameter setting for the GAN, as full grid search optimisation was not feasible. However, its aim was more to identify trends for individual parameters and use this knowledge to inform an improved parameter setting for the GAN in order to try and achieve better results. The performance of the “optimum” GAN will help to understand the robustness of trends in individual parameters and the interdependence between parameters.

Following the results of the GAN hyperparameter experiments, the “optimum” values for each hyperparameter were chosen. This “optimum” GAN setup was then used to train the network again and results were analysed as before, using the FID score and qualitative metrics. Furthermore, the segmentation method described previously was used to assess performance. The number of training epochs was

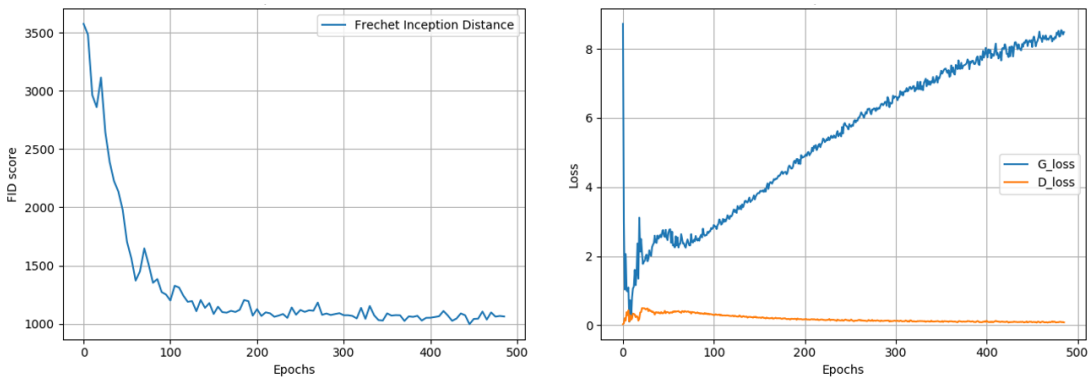


Figure 3.7: GAN training curves for "optimum" GAN training.

increased to 500 to give the GAN a better chance of reaching a higher performance, whilst limiting computational requirements. While there was no guarantee that this would yield a better result, due to the interdependency of parameters, it was nonetheless of interest to try to achieve the best possible result.

"Optimum" GAN Results

The training curves for the optimum run can be found in Figure 3.7. Some example images from the best qualitative epoch can be found in Figure 3.8. These images achieved qualitative rating and classification scores of 4.3 and 0.75 respectively by the author and scores of 4.0 and 0.55 by the experts. A minimum FID score of 995.7 was achieved (995.5 was the minimum FID score recorded during hyperparameter optimisation experiments). Furthermore, the histograms of the distributions of ratios of myocardium to cavity segmentation areas, as well as the separate distributions of cavity and myocardium areas, for real and fake images can be found in Figure 3.9. All three histograms have p-values of $<10^{-2}$, as calculated using the KS-test, meaning there is a statistically significant difference between the real and fake distributions in plots a), b) and c) of Figure 3.9.

"Optimum" GAN Discussion

The employed hyperparameter optimisation strategy is a sub-optimal strategy for finding an optimum GAN setup, as the multidimensional nature of the optimisation problem is being ignored. Only a full grid search would provide a clear insight into

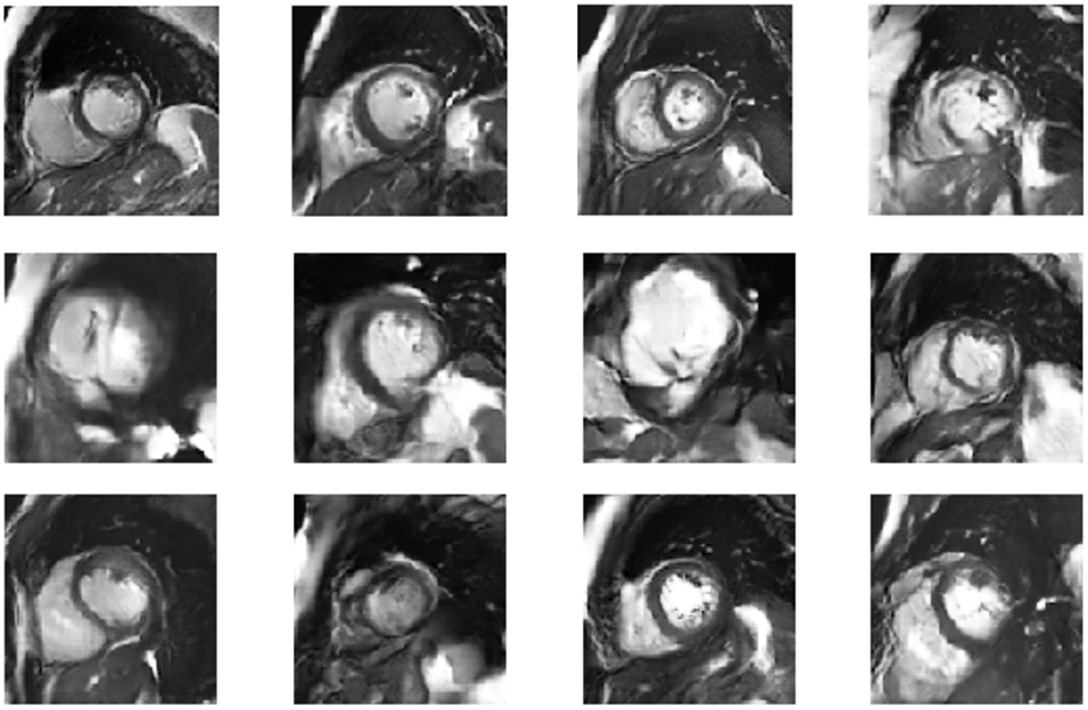


Figure 3.8: Example image results from "optimum" GAN.

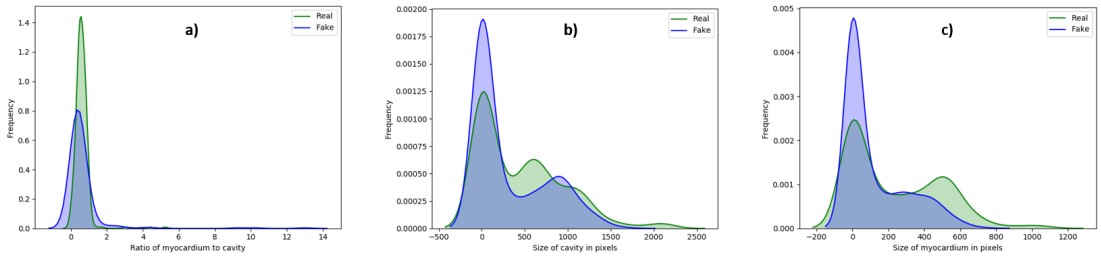


Figure 3.9: a) Distributions of ratios of myocardium to cavity segmentation areas for real and fake images in "optimum" GAN. b) Distributions of cavity segmentation areas for real and fake images in "optimum" GAN. c) Distributions of myocardium segmentation areas for real and fake images in "optimum" GAN. All p-values are $<10^{-2}$ calculated using the KS-test, indicating a statistically significant difference between the distributions.

where an actual performance optimum might lie, however, as previously discussed, this is computationally infeasible. Whilst alternative methods such as Bayesian Optimization could have been used, in this case the robustness of the DCGAN setup was tested against changes in individual parameters and general trends identified. Based on this an "optimum" GAN was built to see whether the previously best seen performance can be improved upon.

Based on the results, it seems that the "optimum" GAN has achieved among the

best results seen, confirming that some of the observed trends might be sufficiently robust and independent of other parameter changes. The GAN training curves in Figure 3.7 are an example of the low level of interpretability that GAN training curves often yield. Other experiments, producing similarly good outputs, have shown very different training curves, depicting high levels of fluctuations in both losses. The FID loss agrees with the previously described evolution of GAN outputs and matches that of many hyperparameter experiments. In the first 100 epochs, gradually improving outputs can be observed, moving from noise to blobs and eventually to rough outlines of cardiac shapes. Following that, the anatomical correctness improves and high-frequency detail gets added as the FID further improves. Though GANs, unlike networks which use converging optimisation functions, can frequently and unpredictably bounce between high quality and low quality outputs, most of the experiments have shown gradually improving outputs.

When samples of the generated and real distributions are qualitatively assessed, the GAN results display a level of diversity that is comparable to that of the real data set. This assessment has however not been done quantitatively across a statistically significant size of samples. Going forward, to better enforce diversity in the generated data set and to ensure that mode collapse has not occurred, several strategies could be employed. These include the use of multiple GANs, though this may come at a high computational cost, methods that minimize entropy to ensure diversity such as Energy-based GAN [241] or methods that add diversity inducing terms to the loss functions such as DRAGAN [102] or MSGAN [242]. Another method by [243] introduces the Pluggable Diversity Penalty Module (PDPM). This centers around the idea that if two latent space vectors are different from one another they should also produce different outputs in image space. If this is not the case then mode collapse is highly likely. To ensure that this diversity is enforced, the PDPM works by comparing the similarity of image features of two images in the discriminator with the similarity of their latent vectors in the generator. If the latent vectors have low similarity but are mapped to images

with high similarity features, then the generator gets penalized. This will help the generator reduce the risk of mode collapse.

As discussed in section 3.3.3, a pre-trained short-axis segmentation network was applied to sets of real and fake images in order to provide an additional performance evaluation tool. The segmenter was pre-trained on short-axis scans from the UK Biobank data set [204]. Therefore, it was not expected to work as well on the Kaggle data as it has done on the UKB data. However, absolute scores are of less interest in this case than the relative performance between the real and fake data sets. As there is no registration between the data sets, the only way to compare the performance of the sets was to examine their segmentations individually.

Upon testing the segmentation network, it was observed that the network would usually output one out of three different types of results. It would either fully segment the relevant structure with a high degree of accuracy, or only segment parts of the relevant structures, or miss them out altogether and pick out nothing at all. Very rarely would the network pick out wrong structures and assume them to be its actual targets. Therefore, using the sum of pixels of the individual segmentation labels could be used as a way of comparing the performance of the segmenter on the data sets and hence the realism of the fake data set itself. The similarity between the distributions of pixel numbers for each segmentation label per image would enable a comparison between data sets. Furthermore, to account for the variety in size of hearts, the ratio of the LV myocardium and LV cavity labels was calculated and plotted. These distributions, as seen in Figure 3.9, seem match to a high degree, however, each set of distributions produces a p-value of $<10^{-2}$ when applying the Kolmogorov-Smirnov test to quantify the similarity between the distributions. This would indicate that real and fake distributions are being distinguished. An explanation for this could be related to the fact that whilst GAN output images can have a realistic appearance this does not necessarily mean that the cardiac geometry is equally well approximated. Therefore the sub-optimal geometry might have had a contributing factor to the difference in segmentation performance between real and generated sets. Another reason could be that the

generated GAN samples that were used for the analysis only covered a part of the real distribution, thereby accounting for the difference in the histograms. Lastly, as previously mentioned, the p-values calculated using the KS-test become less reliable when the number of samples is large.

There are several assumptions made in this segmentation approach. Firstly, it is assumed that the segmentation performance is directly correlated to the quality and realness of the generated images. By analysing example images and their segmentations, it was found that this is roughly true, however, the sample size was not large enough to make this observation statistically significant and be able to confirm this assumption to be true across the data set.

Secondly, given that the segmenter was in fact trained on data from the UKB data set, the assumption has to be made that it will be able to segment the Kaggle data set. Although, as mentioned, absolute segmentation performance is not the main concern, it is still a potential source of error. Lastly, given the large variety of image appearances, problems can arise when comparing distributions as a whole. For example, an image with a large heart can be badly segmented, but produce a sum of pixels that would be in the norm for a smaller sized heart image, and vice versa. Using a distribution as a form of comparison can mask a lot of these underlying errors. Overall, this makes the segmentation metric much weaker as a method of performance evaluation of the GAN than the qualitative analysis and FID scores.

3.4 VAEs for Capturing Cardiac Structure

3.4.1 Introduction

As a comparison to the GAN, a VAE was trained to reconstruct cardiac frames. An architecture similar in structure to the GAN generator was hereby used as the basic VAE encoder and decoder. Firstly, a hyperparameter optimisation was performed to test the network's robustness and identify trends in the VAE setup that might help improve performance. The derived "optimum" VAE was then trained and evaluated and its latent space was analysed to visualise structural modes in the data.

3.4.2 VAE Structure

The basic VAE structure consists of an encoder and a decoder which are combined through a variational latent space, as explained in chapter 2. Both the encoder and decoder follow a similar structure to the discriminator and generator of the GAN in Figure 3.2. The encoder consists of several layers of 2D convolutions, followed by batch normalisations, leaky ReLU activation layers and max pooling operations. The decoder works in reverse by using upsampling layers, followed by convolutions, batch normalisations and ReLU activations, before passing the output through a final convolutional layer with a single filter and a sigmoid activation to generate the output image.

The VAE latent space is composed of two vectors to estimate the mean and variance of the multivariate Gaussian. Each vector has a base length of 128. This is a compromise between values in the literature and computational efficiency and will be a primary hyperparameter to be investigated later. The network uses a loss function that combines a reconstruction loss with a latent space KL divergence loss. While MSE loss is occasionally used in the literature, binary cross entropy is a more common choice and was hence used as the reconstruction loss in this case. The weighting of the losses, also known as the β value, is another important hyperparameter in VAEs and represents the compromise between reconstruction accuracy and latent space regularisation. Whilst the original VAE featured a 1:1

weighting ratio between loss functions, applications in the biomedical imaging field commonly use using higher weightings for the reconstruction loss. Hence a base weighting of 10^{-3} was used between latent loss and reconstruction loss. Similar to the GAN, the VAE also used the Adam optimizer and a learning rate of 10^{-4}.

3.4.3 VAE Hyperparameter Optimization

Optimisation Method

As with the GAN, hyperparameter optimization was an important step for the VAE. However, the assessment of VAE performance was significantly more straightforward than for the GAN, given that ground truth data was available and the functional form of the latent space was set as a normal distribution. The evaluation of each parameter setting was hence directly based on the network validation loss and the minimum FID score, without taking qualitative metrics into consideration. The method of optimisation was again similar to the GAN. Instead of performing infeasible extensive grid searches of all parameter combinations, individual parameters were analysed individually with a range of values from which the best parameter setting was chosen as the "optimum". From this, effects of specific parameter changes on the performance of the VAE could be inferred. The parameters and the value ranges that were investigated are given in table 3.2.

Each network was trained for 300 epochs, as this has been observed to allow sufficient time for networks to converge to an optimum point. For each experiment, the point of minimum validation loss was taken as the optimum point. Based on this FID scores were calculated. The same number of training and validation images that were used for the GAN experiments were also used for VAE experiments.

VAE Hyperparameter Optimization Experiments

Similar to GAN experiments in the previous section, VAE hyperparameter evaluation was done by changing a single parameter at a time with all other parameters taking the value of the baseline setup. VAE hyperparameter experiments were evaluated based on the training and validation loss of the experiments, as well as the minimum

Table 3.2: VAE hyperparameters with associated value ranges. Baseline values for each parameter are indicated in bold.

Parameters	**Value Ranges**
Number of Layers	[2, 3, **4**, 5, 6]
Number of Convolutions	[**1**, 2, 3]
Filter Scaling Factor:	[1, 2, **3**]
Latent Space Dimension	[8, 16, **32**, 64, 128, 256, 512]
Learning Rate	[10^{-2}, 5×10^{-3}, 10^{-3}, 5×10^{-4}, 10^{-4}, 5×10^{-5}, 10^{-5}, 5×10^{-6}, 10^{-6}]
Loss Weightings	[1, 2×10^{-1} , 10^{-1} , 2×10^{-2}, 10^{-2}, 2×10^{-3}, 10^{-3}, 2×10^{-4}, 10^{-4}]

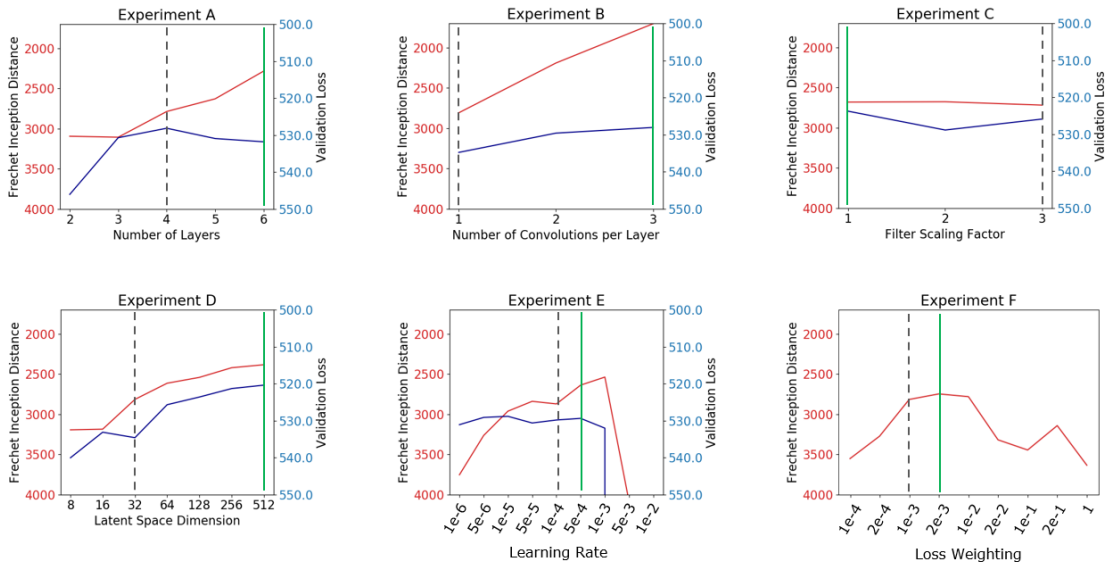
FID score computed on validation images. The hyperparameters and associated ranges are given in table 3.2 with baseline values indicated in bold.

VAE Hyperparameter Optimisation Results

Results of the VAE hyperparameter optimisation can be found in Figure 3.10. Experiments A-F hereby covered the hyperparameters listed in table 3.1 in the following order: Number of Layers, Number of Convolutions per Layer, Filter Scaling Factor, Latent Space Dimension, Learning Rate and Loss Weighting. Red lines indicate the minimum FID scores, whilst blue lines indicate the minimum validation loss. Vertical dashed grey lines give the base line value for each parameter prior to optimisation and solid green lines give the chosen optimum value. Optimum values are summarised in the table in Figure 3.10.

VAE Hyperparameter Optimisation Discussion

As with the GAN hyperparameter experiments, each experiment used a base set of parameters, but progressively altered a single parameter through a range of values. Given the availability of ground truth data, no qualitative metrics needed to be employed. Hence the figures only display the minimum FID scores achieved by each individual parameter setting, as well as the minimum validation loss. Lower FID scores and validation losses indicate a better performance.



Hyperparameter	Optimum Value	Hyperparameter	Optimum Value
Number of Layers	6	Latent Space Dimensions	512
Number of Convolutions per Layer	3	Learning Rate	5e-4
Filter Scaling Factor	1	Loss Weighting	2e-3

Figure 3.10: Results of VAE hyperparameter experiments A-F. The red line indicates the FID score. The dark blue line represents the validation loss. The vertical dashed grey line indicates the base line value for each parameter prior to optimisation, whilst the vertical solid green line indicates the chosen optimum value. Optimum values are also summarised in the attached table.

The agreement between the metrics across the experiments is mostly good. In experiment A, FID scores seem to indicate that a higher number of layers is favourable. Validation losses do not necessarily agree with that trend and display a minimum loss at 4 layers, with a marginal increase in losses for higher numbers of layers. The absolute difference in loss value, however, is almost negligible. According to FID scores, the results seem to suggest that even better scores might be achieved with additional layers. This proved not to be the case though. A follow-up test was run with 7 layers, at which an FID score of 2538 and a minimum validation loss of 536.0 were achieved, which are significantly higher than the scores of the 6 layer version. Reasons for that are likely to be related to the filter dimensions at the bottleneck. After 7 maxpooling layers, the original 128x128 dimension has been reduced to a 1x1 length. This means that any spatial encoding which, though

coarsely, is still existent in a 2x2 image, has now been lost entirely, making a reconstruction more difficult. An optimum value of 6 layers was chosen in the end.

Experiment B shows strong agreement between metrics, both suggesting that more convolutions might help improve network performance. This stands in contrast to observations made during the GAN hyperparameter experiments which showed no strong agreement between metrics on the number of convolutions for the generator, and even showing an inverse correlation in the discriminator. While the trend suggests that an even larger number of convolutions per layer could potentially result in an even higher level of performance, the investigation of more than 3 convolutions per layer was found to be computationally infeasible. Generally speaking though, if resources allow, optimisation value ranges should be extended if the extreme values of a range reach an optimum value within the given range.

Results from experiment C show good agreement and seem to indicate that the number of filters has little to no effect on VAE performance. Mild improvements for a value of 1 hereby dictated the choice of parameter.

Experiment D indicates that network performance correlates well with the dimension of the latent space. This is to be expected, as a more restrictive VAE bottleneck leads to larger information loss. Reconstructing a high dimensional output from a lower dimensional input becomes harder the smaller the input dimensions become, as the variation in the data becomes more difficult to capture. It is of course precisely this restriction that forces the network to compress data in a more efficient low dimensional space, pushing it to capture the largest variations and modes in the data. However, this compression forms a trade-off with network performance. For reconstruction purposes, a value of 512 was chosen as the optimum.

Results from experiment E show that for learning rates below 10^{-5}, the FID score seems to indicate sub-optimal network performance. The validation loss, however, disagrees with that and indicates comparable performance for most of the lower learning rates ($<10^{-2}$). The training curves of the VAE also indicate that the computational time frame of 300 epochs was sufficient even for the smallest learning rates for the VAE to reach a point of optimum training and overfitting. For

learning rates above 10^{-5}, both FID score and validation loss indicate good results, with FID scores significantly favouring higher learning rates. Yet for learning rates above 10^{-2}, the network stops producing sensible results and instead just outputs noise. Both FID scores and validation loss accordingly indicate poor network performance. The network seems to be unable to converge to a local minimum and instead just bounces around the error space. Overall it can be said that, as expected, the learning rate has a strong influence on the performance of the VAE. A value of 5×10^{-4} was chosen as the optimum.

Experiment F centered around changing the respective weighting of the KL and reconstruction loss in the VAE. This is effectively the same as changing the β parameter in a β-VAE. The validation loss was not displayed in the graph for this experiment as the result would not have been interpretable. The validation loss is a function of the loss weighting and changes in absolute value as the loss weighting changes. Therefore, including it as a total absolute value is inconsequential. Instead, a qualitative analysis, as previously used for GAN hyperparameter experiments, was applied to analyse the results of this experiment. Practically the β parameter changes the relative weighting of the KL loss while the reconstruction loss is continuously weighted at 1. The FID score shows that there is a potential optimum in the weighting between 10^{-2} and 10^{-4}. However, the qualitative analysis showed that this is more nuanced. High values of β, i.e. a high weighting of the KL loss, seem to produce very blurry images, with anything larger than β=2×10^{-2} producing noisy outputs. Furthermore, even though the FID score seems to indicate relatively low performance for β values $<10^{-4}$, qualitatively, these produce much more realistic outputs than those of very high β values. This is another example of how FID scores can be misleading with regards to the qualitative performance of generative networks. An optimum value of 2×10^{-3} was chosen. A summary of all optimum parameters is given in the table in Figure 3.10.

3.4.4 "Optimum" VAE

"Optimum" VAE Experiment

The optimum values for each parameter were used to re-train an "optimum" VAE with the aim of achieving better network performance. The same training parameters, as used for the hyperparameter experiments, were hereby applied. Network training was stopped according to an early stopping strategy using a persistent increase in validation loss over 5 continuous epochs as the trigger. The epoch with the lowest validation loss was then used for testing. During test time FID scores and test loss were computed. To showcase the VAE as a generative network that has captured cardiac structure, random latent space inputs drawn from a Gaussian distribution were used as the decoder input to produce random output samples.

"Optimum" VAE Results

The "Optimum" VAE achieved a minimum FID score of 1700 with a test loss of 518.4. Example VAE image outputs following random Gaussian latent inputs can be found in Figure 3.11.

Discussion & Comparison of "Optimum" VAE and GAN

On average the generated VAE images display a much higher level of anatomical correctness than their GAN counterparts. This aids with their realistic appearance. However, contrary to GAN images, the VAE outputs often show a lot of image blurring, making them look like a downsampled or blurred version of real images. This blurring effect has been described in chapter 2. If an underlying application requires images to "look" real, then this blurriness makes them easily identifiable as fakes. However, if realistic geometry is required, then VAE outputs might be superior to those of GANs.

To compare the performance of GANs and VAEs at capturing cardiac structure from CMR data, both the quantitative and qualitative metrics were examined. Based on the FID score, GAN results clearly outperformed VAE results. The lowest FID score recorded for VAEs was 1701 (a minimum score of 1732 was recorded

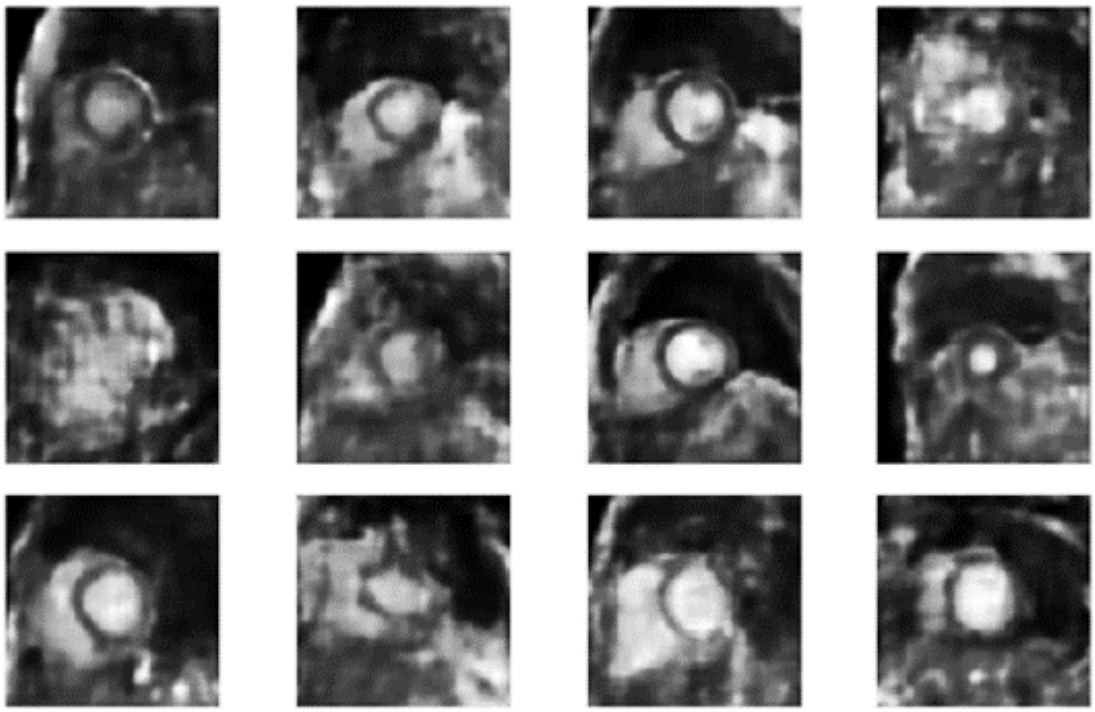


Figure 3.11: Example outputs from "optimum" VAE random Gaussian inputs.

during hyperparameter optimisation). This score was outperformed by around 70% of hyperparameter settings in the GAN experiments. The lowest GAN FID score was placed at 995.5 (995.7 for “optimum” GAN). The FID score seems to be very sensitive to high frequency information in the images, which the GANs were much better at.

To compare the qualitative performance between both network types, the optimum VAE underwent the same qualitative assessment of rapid classification and rating as the GAN networks previously. The author’s qualitative rating and classification results were 3.5 and 0.97 respectively and the expert results were 3.9 and 0.91.

Compared to the optimum GAN scores, the VAE results do similarly well, and only slightly worse, in terms of qualitative rating scores. This is probably related to the fact that anatomical accuracy of VAE results is high. However, when comparing classification scores, a much larger difference in performance can be seen. With an average classification accuracy of 94%, VAE results can easily be classified as fakes, compared to an average GAN classification accuracy score of 65%. A summary of quantitative and qualitative scores for both the GAN and VAE are given in table 3.3.

Table 3.3: Comparison of quantitative and qualitative results for optimum GAN and VAE performance in capturing cardiac structure. Note that lower Minimum FID scores, higher Qualitative Rating Scores and lower "Fake" Classification scores indicate better network performance.

Metric	GAN	VAE
Minimum FID Score	996	1700
Average Qualitative Rating Score (/5)	4.2	3.4
Average "Fake" Classification Score in % (/100)	65	97

3.4.5 VAE Latent Space Analysis

VAE Latent Space Introduction

As outlined in chapter 2, VAEs, in contrast to autoencoders, have a regularised latent space which should theoretically ensure continuity and completeness. The latent space vectors should therefore capture a distribution that allows for interpretable dimensions. In other words, when moving across the latent space, incremental modal changes in the observed output should occur. In order to investigate this theory with regards to cardiac data and to understand structural modes among the data set, the VAE latent space was studied.

Latent Space Experiments

Once fully trained the decoder of the VAE was used for synthetic data generation. Hereby one-hot encoded latent space vectors were used as inputs to visualise the modes captured by each latent space vector element. To visualise the impact of each vector element on the generated output, the one-hot encoded input was changed through a range of values spanning between the first standard deviations either side away from the mean of the latent space distribution.

Even though the KL loss is given a much lower weighting than the reconstruction loss, the latent space was assumed to approximate a Gaussian in nature. Therefore, taking a standard deviation either side as the limits was deemed applicable. It must be noted that the absolute values for the latent space inputs were rather irrelevant, as the interest of this task was to extract the modal change that a specific vector element is responsible for, not the appearance for specific latent space input values.

Latent space experiments were first run with the base line VAE to examine different modes of change. This had a latent space dimension of 32 which was deemed as a good compromise between reconstruction accuracy and dimensionality reduction. Later the latent space dimension was changed to 24 and then to 16 to investigate its effect on capturing data modes.

Additionally the base line VAE was re-trained, but with a high value for β parameter, i.e. the loss weighting.

Latent Space Results

Results for VAE latent space experiments can be found in Figures 3.12, 3.13 and 3.14. Figure 3.12 depicts several modes of change that were extracted from decoder outputs following one-hot encoded inputs. These only activated a single latent space vector element at a time with values taken from a sequence of equidistantly spaced values in the range of [-σ, σ], where σ is the standard deviation of the latent space distribution during test time.

Figures 3.13 and 3.14 show outputs from VAEs re-trained with latent space dimensions of 24 and 16 respectively whilst Figure 3.15 shows outputs from re-training the VAE with a high β value.

Latent Space Discussion

The analysis of the latent space experiment results in Figure 3.12 clearly highlights the dependence of the VAE output on the latent vector inputs. The value of individual vector points seems to correlate with modal changes in the output which include features such as LV size, RV size, heart rotation, scaling or image contrast. While these changes are easy to observe, it is not always straightforward to describe and identify exactly what type of mode change is occurring. The reason for this is that most of the changes are not disentangled from each other and seldom governed by individual vector points. Instead, each vector point seems to correspond to a composite mode change that includes a combination of the aforementioned feature changes. Therefore, assigning descriptive labels of mode changes to particular vector indices is not always straightforward. Nonetheless, for many vector points,

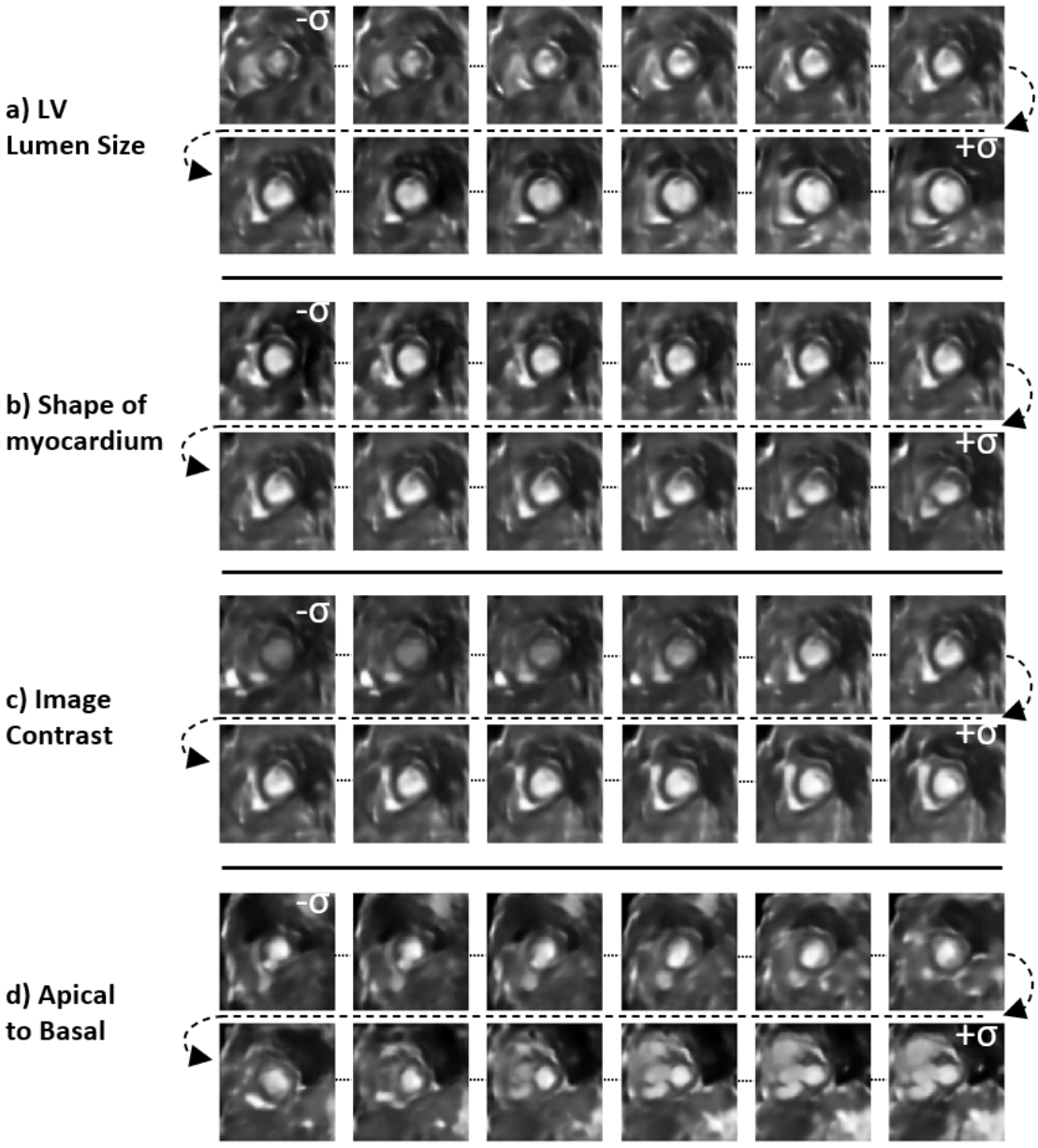


Figure 3.12: VAE outputs given one-hot encoded latent space inputs whereby only single latent space vector elements were activated at a time with equidistantly spaced values taken from a range of [-σ, σ], where σ is the standard deviation of the latent space distribution during test time. The outputs demonstrate different modes of change for identifiable anatomical features, namely LV Lumen Size, Myocardium Shape, Image Contrast and Apical to Basal Position, related to specific latent space vector elements.

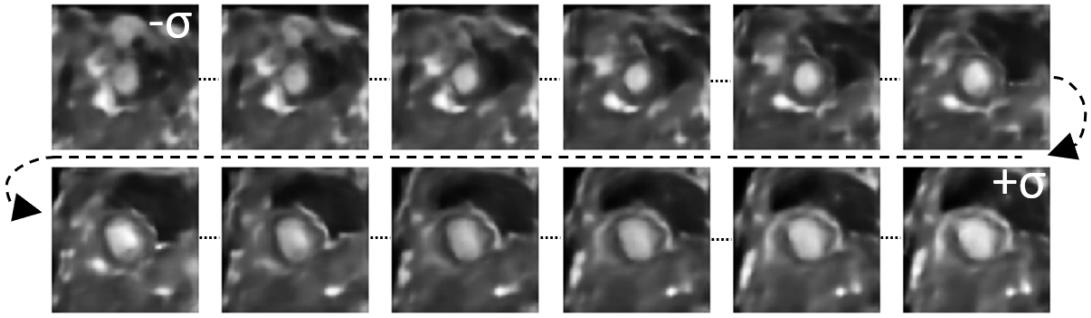


Figure 3.13: Example VAE outputs given sequence of one-hot encoded latent space activations for vector element values in range [-σ, σ], where σ is the standard deviation of the latent space distribution during test time, when latent space dimension is of size 24.

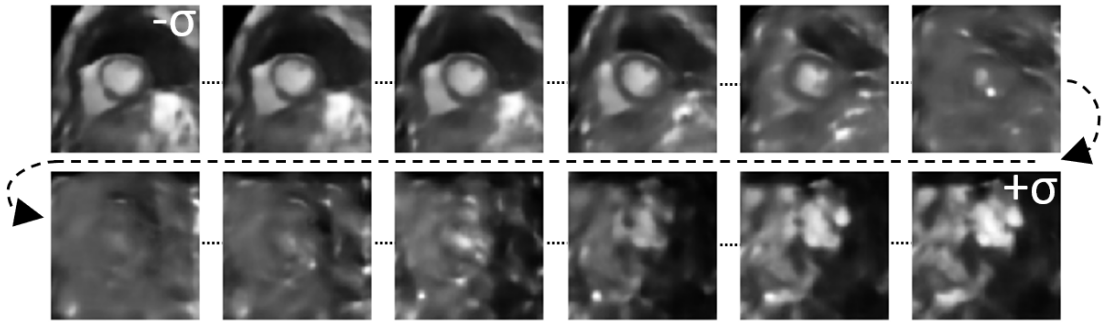


Figure 3.14: Example VAE outputs given sequence of one-hot encoded latent space activations for vector element values in range [-σ, σ], where σ is the standard deviation of the latent space distribution during test time, when latent space dimension is of size 16.

the mode changes do seem to be dominated by one or two major modal changes. In general it can therefore be said that VAEs are able to capture the variation in CMR data sets and that these have a high degree of correlation with human anatomical description of the data.

At this point it is worth noting that with the network configuration that was used in this work the latent space could only be analysed passively by visualising the effect that changes in specific vector elements have on the generated output. Actively disentangling and linking the latent space to specific features was however not possible. Alternatives such as InfoVAEs [182] and InfoGANs [124] which use a latent code c to target significant semantic features and allow for more active control of the latent output could provide a better avenue for this in future work.

It is interesting to note that when using a latent space of length 32, around 40% of the latent space vector points are "dead" nodes, meaning that a change in their input has no effect on the VAE output. This was expected and highlights the trade-off that must often be considered when choosing the dimension of the latent space. A lower number of dimensions forces the network to compress the data and make efficient use of the available capacity in the network to represent the data. The number of dead nodes should theoretically decrease and the variation between the nodes increase. When re-training the network with latent spaces of size 24 and 16, a significant reduction in dead nodes to 25% and 0% of the nodes respectively could be observed. Furthermore, the modes seemed to become more comprehensive, as seen in Figures 3.13 and 3.14. It should be noted though that dead nodes are not necessarily redundant. Experiments to investigate and proof the potential redundancy of dead nodes were not conducted.

These figures, however, also highlight the drawback of reducing the latent space dimension which is that it also impacts the network's ability to accurately reconstruct the input. The outputs become blurrier and of lower visual quality, which in effect means that assessing the nature of the modal changes also becomes more difficult.

To better disentangle the latent space, the β coefficient in the β-VAE setup needs to be increased. This gives a higher weighting to the KL loss in the overall VAE loss function and therefore ensures a more Gaussian latent space. However, similarly to reducing the size of the latent space, this reduces the reconstruction accuracy of the images, thereby making it in fact more difficult to visually assess the disentanglement of the latent space, even though it might exist. Figure 3.15 demonstrates the effect of using a high β value.

3.4.6 New Data Generation

It has been shown that GANs and VAEs can in fact be used to generate synthetic CMR data that comes close to real data in terms of image quality. This means that generative networks such as these can be used to create new synthetic CMR data sets which could be used as training sets for further CMR deep learning

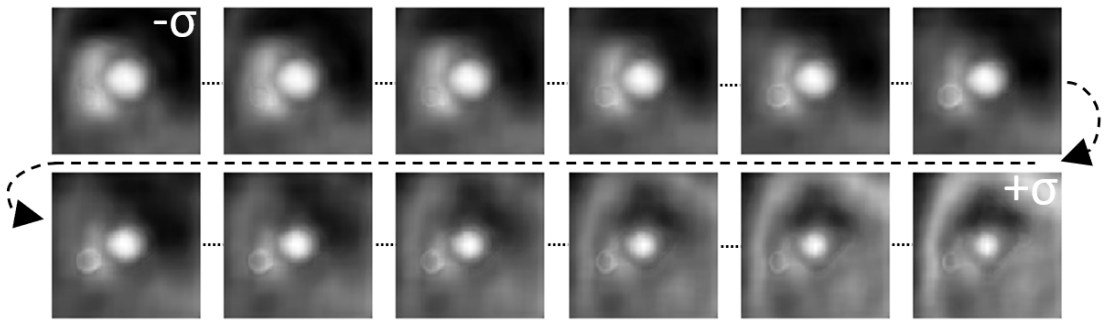


Figure 3.15: Example VAE outputs given sequence of one-hot encoded latent space activations for vector element values in range [-σ, σ], where σ is the standard deviation of the latent space distribution during test time, when latent space has a high β weighting.

tasks. This would help with the scarcity of CMR training data sets in the research world. Further optimisation of architectures and hyperparameters are likely to produce images of even higher quality.

In general, GANs have outperformed VAEs with regards to realness of appearance, however, VAEs offer a higher degree of control in the generative process. GANs do not use a regularized latent input space, meaning that there is less control over the output that is created than in VAEs. Following VAE training and identification of latent space modes, the decoder can be used to generate specific sub-populations of synthetic data. For example, if a latent space vector point has been identified to be responsible for controlling LV cavity size, this insight can be used to generate synthetic data belonging to this specific region in the latent space, i.e a subgroup of the data distribution where hearts have larger LV cavities. Figure 3.16 depicts such an image generation process. Here, random input vectors have been fed to the latent space, except for a single vector element which received an input of 0 and later an input of +2, with all other values remaining the same. This vector element has previously been found to be associated with LV cavity size, or in broader terms with heart size in general, since the disentanglement was not perfect. It can clearly be seen in the Figure that using a high activation of this latent vector element resulted in an output of larger hearts (VAE outputs B in the Figure), compared to using a zero activation (VAE outputs A in the Figure). This shows that by understanding the latent space of the VAE, the image generation process can be steered.

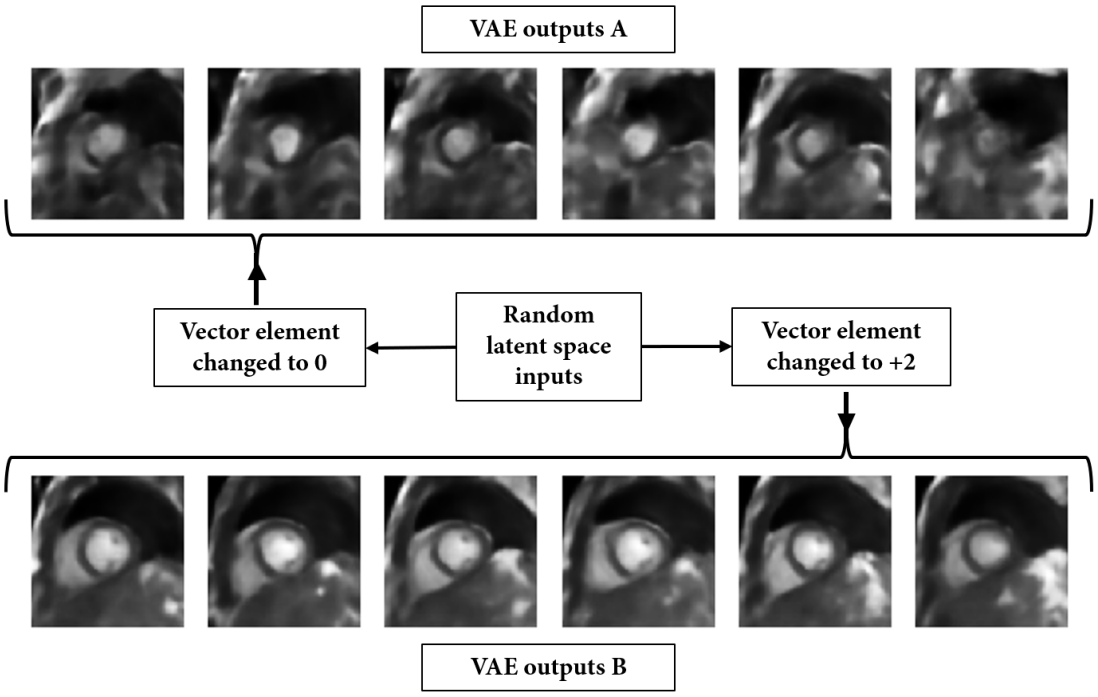


Figure 3.16: VAE output results of changing single element in latent space that is known to be associated with LV cavity size. The same random latent space inputs were augmented by changing the same element to 0 to produce VAE outputs A and then to +2 to produce VAE outputs B.

3.5 Conclusion

Generative networks, both in the implicit form of a GAN and the explicit form of a VAE, have been shown to be able to successfully capture CMR structure. GANs in particular have been able to generate images that are able to fool experts into thinking they are real.

Using hyperparameter optimization has provided an insight into the effect of certain hyperparameters on network performance for both GANs and VAEs, and identified trends. These need to be treated carefully though as the interdependence between parameters is not negligible.

Furthermore VAE latent spaces have been analysed and it was shown that certain latent vector elements correspond to anatomically identifiable modes in the data. These can be used to steer data CMR data generation.

Limitations of both GANs and VAEs have been outlined in the chapter and largely relate to either anatomical incorrectness or blurring effects. As a next step it would therefore be useful to combine the power of control and anatomical correctness from VAEs with the sharp and realistic image features from GANs. Networks such as adversarial VAEs and VAE-GANs would form a natural extension to this work.

Overall the work formed in this chapter can be used as a basis for generating new synthetic CMR image data that can be utilized in further DL training applications, eg. providing extra training data augmentations for image segmentation tasks. The ability to both extract modes from data sets and manipulate the image generation process accordingly also enables the generation of sub-population specific data sets for the use of more specific DL applications.

The key contributions of this chapter are:

- Development of an optimised Generative Adversarial Network (GAN) and Variational Autoencoder (VAE) for learning the appearance of Cardiac Magnetic Resonance (CMR) frames at different positions in the cardiac sequence

and at varying positional levels along the short axis of the heart to be used for the generation of synthetic CMR data.

- Identification of modes of change in the CMR data distribution through the analysis of the VAE latent space.
- Generation of specific sub-population synthetic CMR data for further training of cardiac deep learning networks.

4

Cardiac Motion - Healthy and Diseased

Contents

4.1 CGANs for the Prediction of Cardiac Contraction . . 130
4.1.1 Disclaimer . . . 130
4.1.2 Introduction . . . 130
4.1.3 Related Works . . . 132
4.1.4 Methods . . . 133
4.1.5 Experiments . . . 135
4.1.6 Results . . . 136
4.1.7 Discussion . . . 140
4.1.8 Conclusion . . . 146
4.2 Metadata Inclusion . . . 148
4.2.1 Introduction . . . 148
4.2.2 Outline . . . 148
4.2.3 Metadata vs No Metadata . . . 149
4.2.4 Individual vs Joint Network . . . 156
4.2.5 Multiple Metadata Inputs . . . 158
4.3 Conclusion . . . 163

4.1 CGANs for the Prediction of Cardiac Contraction

4.1.1 Disclaimer

The work in this section is based on a publication at the STACOM MICCAI 2019 Workshop: **J. Ossenberg-Engels**, V. Grau *"Conditional Generative Adversarial Networks for the Prediction of Cardiac Contraction from Individual Frames"* In: International Workshop on Statistical Atlases and Computational Models of the Heart. pp. 109–118 (2019) [244].

4.1.2 Introduction

Motivation

As mentioned in the introduction, cardiovascular diseases account for 31% of annual fatalities worldwide, making them the most common cause of death. Reasons for this include the poor understanding of many such diseases, which frequently results in the ineffective treatment of patients leading to sub-optimal clinical outcomes. In particular, the relationship between disease phenotype and clinical outcome is often poorly understood. This is a particular challenge for pathologies such as hypertrophic cardiomyopathy (HCM), which present with high phenotype heterogeneity, with a wide range of structural cardiac abnormalities such as myocardial wall thickening or aortic outflow obstructions. Structural changes such as these can affect the mechanical function of the heart, i.e. cardiac contraction, in a variety of ways. However, the exact relationships are not well understood.

Since it is largely cardiac function that determines clinical outcomes, a better understanding of the relationship between cardiac structure and function could lead to more effective risk estimation, more accurate diagnosis and consequently better treatment of patients.

Conditional GANs

This chapter builds on the study of cardiac structure in chapter 3 by analysing cardiac function in response to cardiac structure. Whilst structure-function relationships

have been studied, this has mostly occurred in isolated approaches typically examining specific global parameters and clinical biomarkers (e.g. cardiac wall thickness/ventricular volumes vs ejection fraction (EF)). Although such biomarkers are clinically established, there is agreement [2, 3, 245] that they are often too coarse for accurate clinical prognosis. To better understand these relationships and to subsequently improve outcome predictions and treatments, it is likely that a more comprehensive approach is necessary which could be achieved using generative methods. In this chapter the foundations for an approach based on conditional Generative Adversarial Networks (cGANs) have been developed. cGANs, as shown by Isola et. al [128], can be used for image transformation tasks and are therefore suitable for transforming one frame of the cardiac sequence into another, thereby modelling the cardiac sequence and the functional behaviour of the heart. Here, the primary interest lies on the transformation of end-diastolic (ED) to end-systolic (ES) frames, which represent the two extreme states in the cardiac cycle.

The choice for focusing on just these frames has several reasons. Firstly, ED and ES are generally speaking the frames that are the most different from one another, representing the starting points of two very distinct physiological processes. Secondly, ED and ES are commonly used in clinical research and therefore well understood. Thirdly it was assumed that while modelling specific pathologies which are defined by changes along the contraction or dilation process might benefit from more frames, modelling a general population would only see marginal benefits. And lastly, modelling the entire sequence exceeded the scope of this work in terms of time resources.

Hypothesis

It is therefore the hypothesis of this chapter that by training a cGAN to transform ED to ES frames healthy cardiac motion can be modelled. Moreover, it is hypothesised that this network is capable of capturing cardiac motion that is specific to individual patient groups. In order to validate these hypotheses three stages of experiments are conducted. Firstly, the network is trained on a data set of healthy volunteers

representative of a heterogeneous population as given in the UK Biobank study [246]. The network is evaluated based on how accurately the ES frame predictions match the acquired images for a test set using image and segmentation similarity metrics (MSE, SSIM & Dice). The assumption is that an accurate prediction of ES frames indicates that the cGAN has captured healthy cardiac motion. In a second stage, the network is re-trained on data from specific healthy patient groups within the UK Biobank and tested on separate test sets from the same and different groups. The assumption is that the cGAN will perform sub-optimally when tested on the latter groups, thereby indicating that it has captured features of the cardiac structure-function relationships that are specific to certain patient groups. As a third experiment this approach is extended to pathological data. The network from the first experiment which has been trained on a healthy population is evaluated on a test set of patients diagnosed with HCM, as well as normal healthy volunteers. It is hypothesised that the network will perform better on healthy volunteers compared to pathological cases, indicating that healthy cardiac motion has in fact been captured by the cGAN.

This method aims to capture a correlation between structural and functional changes, rather than causal relationships, as such are likely to be difficult to isolate and would probably require biophysical models of contraction whereby cardiac properties can be altered and their effects analysed.

4.1.3 Related Works

With regards to HCM, current clinical strategies applied for HCM outcome risk prediction, in particular for the risk of SCD, involve complex models taking into account several clinical parameters from which risk estimations are regressed [247]. Several works have attempted the identification of risk factors for SCD [248][249], e.g. using machine learning with arrhythmic risk markers to identify distinct phenotypes from ECG and echocardiography features [250–252] or investigating the role of outflow tract obstructions [253] and wall motion abnormalities [254] as prognostic tools.

4.1.4 Methods

Conditional GANs

The cGAN used in this chapter is based on the method proposed by Isola et al. [128] Two networks, namely a generator G and a discriminator D, compete in a minimax two-player game, causing them to learn in an adversarial fashion. The generator, a U-Net (Figure 4.1), takes a grayscale ED image as an input and generates a "fake" transformed ES image as its output. The discriminator, which is the decoder part of a PatchGAN with the same structure as the encoder part of G, except also with an additional softmax layer at the end, takes both fake and real ES images and learns to discriminate between them in a supervised manner. The discriminator loss given in Equation 4.1 is used to train both D and G using mini-batch gradient descent and the Adam optimizer.

$$G = \arg\min_{G}\max_{D} \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G), \tag{4.1}$$

where

$$\mathcal{L}_{cGAN}(G, D) = \mathop{\mathbb{E}}_{x,y}[\log D(x, y)] + \mathop{\mathbb{E}}_{x,z}[\log(1 - D(x, G(x, z))] \tag{4.2}$$

In addition to the $\mathcal{L}_{cGAN}$, which is a conditioned version of the GAN loss, Equation 4.1 also incorporates an $L1$ reconstruction loss which forces the transformed image to be close to the ground truth in an $L1$ sense. Hyperparameters and general methodology are based on the Isola method. However, the number of layers and filters in G and D have been adapted following empirical cross-validation. Specific values of these can be found in Figure 4.1. The performance of the cGAN at accurately predicting ES frames is validated by computing the Mean Squared Error (MSE) and Structural Similarity Index (SSIM) between ES predictions and ground truth images. These metrics are computed over the entire image.

Segmentation U-Net

As seen in the workflow diagram of Figure 4.2, following the ED to ES image transformation, the ES prediction and ground truth images are fed into the U-Net segmentation network created by Bai et al. [204], who trained it on data

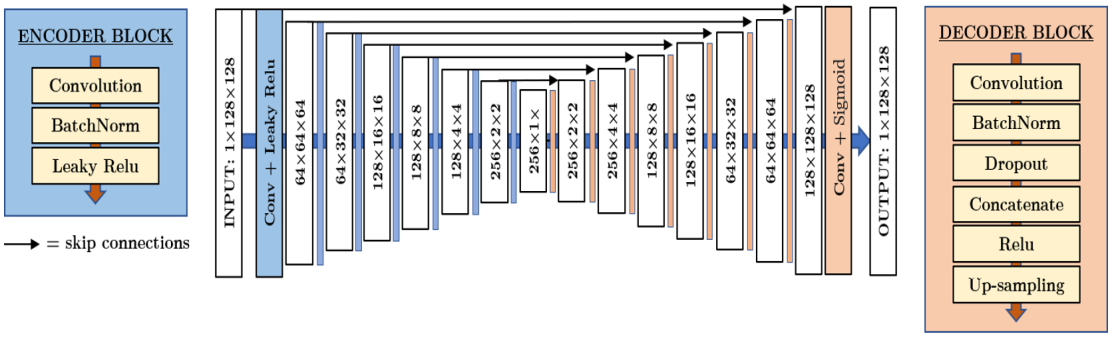


Figure 4.1: cGAN U-Net Generator architecture; Numbers indicate number of filters × image dimensions for each layer. The Discriminator architecture is equivalent to the encoding part of the Generator, followed by a softmax layer at the end.

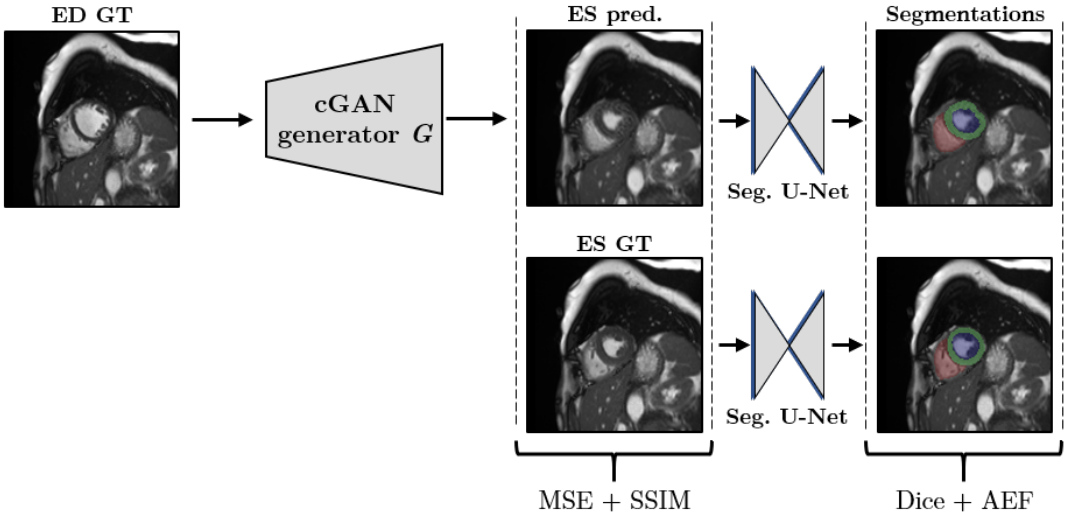


Figure 4.2: Workflow diagram: ED Ground Truth (GT) frames are fed into cGAN generator to produce ES predictions from which MSE and SSIM are computed; Predictions and ES GTs are segmented by U-Net from which Dice and AEF are computed.

from the UK Biobank to segment short-axis frames into a four-label segmentation output, namely left ventricle (LV) cavity, LV myocardium, right ventricle (RV) and background. ED frames are pre-processed in order to fit the orientation and size of the segmentation network input. The binary LV cavity segmentations from predicted and ground truth ES images are compared using Dice coefficients.

4.1.5 Experiments

Healthy cardiac motion prediction

The data for the first two experiments was obtained from the UK Biobank study. The initial experiment was conducted using the data of a heterogeneous group of "healthy" volunteers. As the UK Biobank is designed to represent the general population in terms of the distribution of patient parameters, some patients will develop pathologies during the data acquisition period. However, this fraction is deemed small enough to continue referring to the Biobank data as "generally healthy". Short-axis mid-axial cardiac sequences were selected from ∼12,000 patients from which ED and ES frames were extracted. The images were subsequently cropped with a 128x128 pixel window centred on the LV endocardium, using rough LV automated segmentations included in UK Biobank as guidance. Image intensities were normalised to a [0-1] range. The image pairs were split into independent training and test sets, with no inter-subject overlap, i.e. data from one patient cannot appear in both training and test sets, with ∼33,500 ED-ES image pairs for training and a further ∼4,500 for testing. No image augmentation was applied, given the size of the training set. The cGAN was trained on a 5GB 1060 Nvidia GPU for 200 epochs (∼8hr training time).

Subject group re-training

Following the evaluation of the ES frame prediction for a general population of healthy volunteers, the network was re-trained with data from specific subject groups, differentiated by gender, age and Body Mass Index (BMI), also drawn from the UK Biobank study. For gender, the cGAN was separately re-trained on ∼27,500 ED-ES image pairs from female healthy volunteers and tested on separate data sets of female and male healthy volunteers with ∼4,500 image pairs each. For age, the network was re-trained on ∼5,300 ED-ES image pairs from "young" healthy volunteers (ages 40-45) and tested on separate data sets of young and old (ages 65-70) healthy volunteers with ∼700 image pairs each. Lastly, the network was trained on ∼6,000 ED-ES image pairs from volunteers with low BMIs (BMI<22)

and tested on separate sets of subjects with low and high BMIs (BMI>38) with ∼800 image pairs each. The training parameters, except for the training time, were the same as for the first experiment.

Pathological cardiac motion in HCM patients

The network trained in the first experiment was re-applied to data from [255]. This study consists of cine CMR scans of 55 HCM phenotype positive, 18 HCM phenotype negative but genotype positive and 37 normal volunteers (NV). Only the phenotype positive and normal volunteers were considered for testing. Pre-processing steps including histogram matching, LV centering and cropping were applied to the data in order to match the UK Biobank data in appearance. Short axis mid-axial sequences were selected from 34 and 22 HCM patients and normal volunteers respectively from which 102 and 66 mid-axial ED-ES pairs were extracted and used for testing.

4.1.6 Results

cGAN ES frame predictions

Example results of cGAN ES frame predictions when trained on a general healthy population and their corresponding segmentations can be seen in Figure 4.3 and 4.4. Mean MSE and SSIM scores between the ES ground truth and ES prediction were 0.0026 ±0.0013 and 0.89 ±0.032 respectively. The mean Dice score between the LV myocardial segmentations of ES ground truth and predictions for the general population was 0.89 ±0.059.

Subject group differences

The analysed subject groups were: female/male, young/old and low/high BMI. For each pair the network was trained on the former group and tested on separate validation sets from both groups. MSE and SSIM values have been computed using the ES ground truth images and ES predictions for each patient group. Subsequently, all images have been segmented and Dice coefficients of the segmentations have been computed. Using the LV cavity segmentations, area ejection fractions (AEF) were computed according to $AEF = (EDA - ESA)/EDA$, where EDA is the

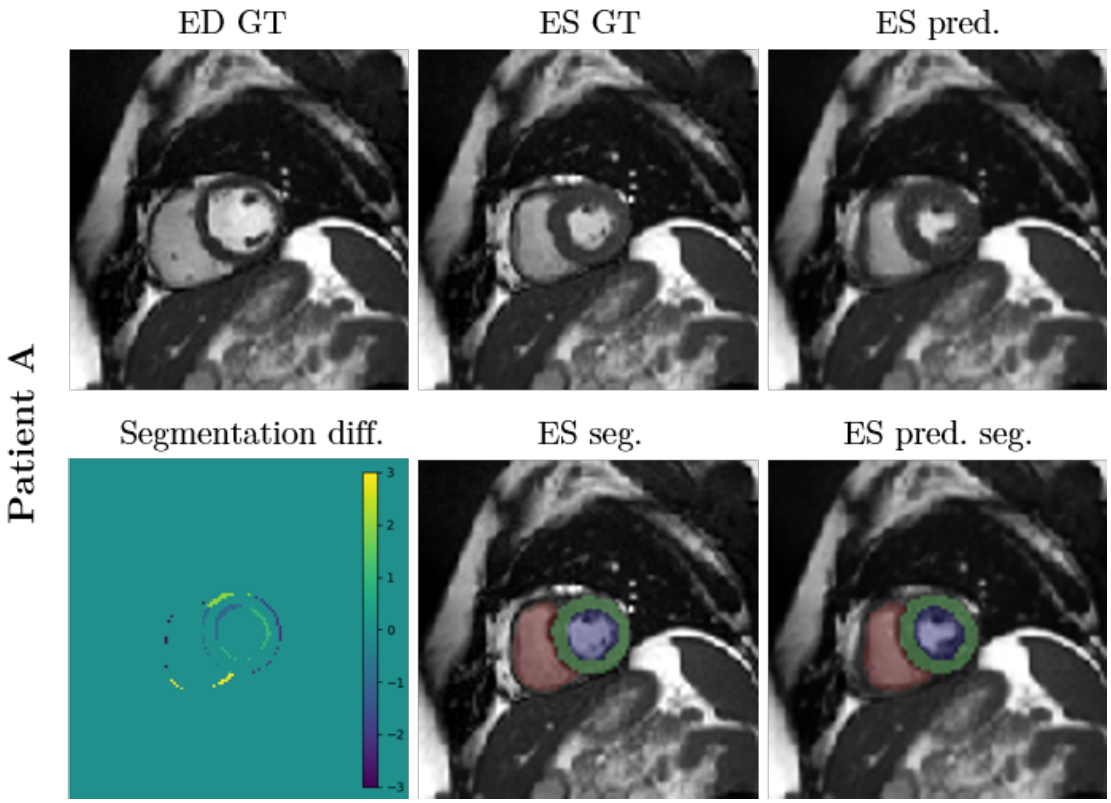


Figure 4.3: cGAN prediction of ES frame for Patient A, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0035, 0.89 & 0.91] respectively.

end-diastolic area and ESA the end-systolic area. The mean percentage differences between the AEF of ground truth images and predictions were computed, as seen in Table 4.1. The SSIM and Dice scores of the individual patient groups have been plotted on normalised histograms to display frequency distributions, as seen in Figure 4.5, which also shows p-values between the distributions calculated using the Kolmogorov-Smirnov (KS) test.

HCM patients vs normal volunteers

SSIM and MSE scores between the predicted ES frame and the ground truth ES images have been computed for NV and HCM cases. The segmentations have been compared using Dice and AEF scores and together with SSIM and MSE are displayed in Table 4.1. Furthermore the histograms of SSIM and Dice scores of the two groups have been plotted and can be seen in Figure 4.5 which also show

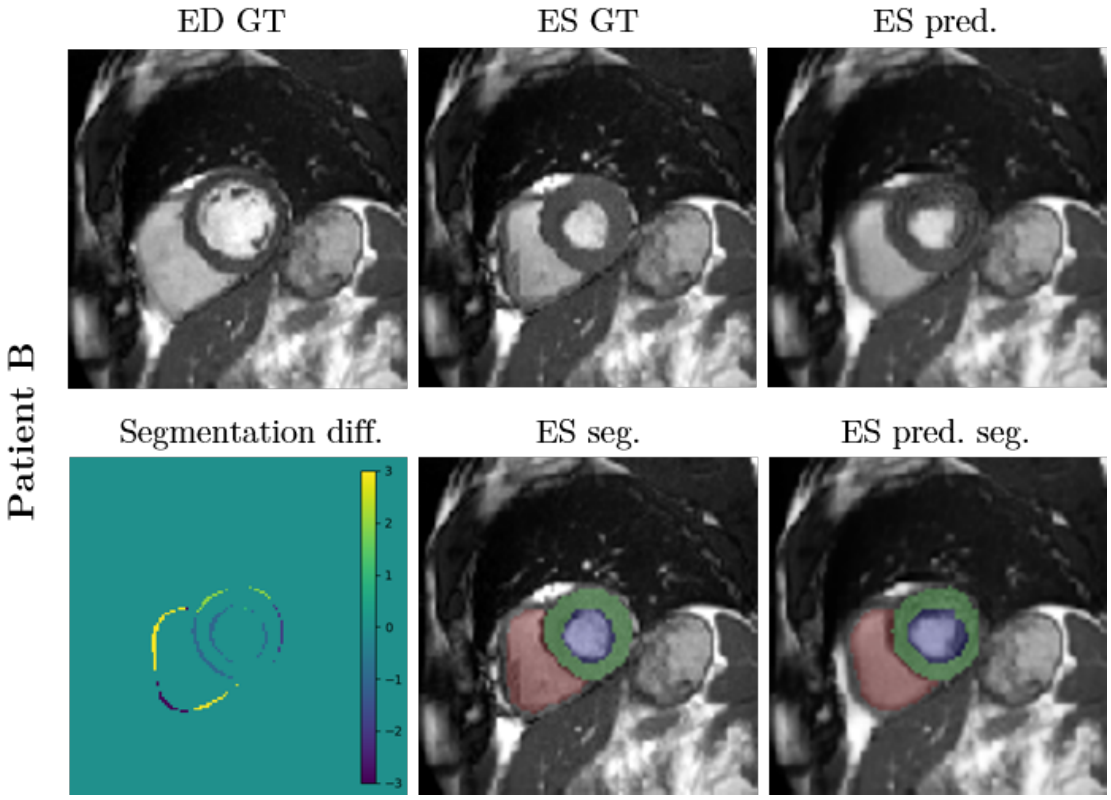


Figure 4.4: cGAN prediction of ES frame for Patient B, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0047, 0.85 & 0.94] respectively.

Table 4.1: MSE (in 10^{-3}s), SSIM, Dice scores and AEF differences for patient groups. Note that lower MSE and AEF, and higher SSIM and Dice, indicate better predictions.

Subject Group	**MSE score**	**SSIM score**	**Dice score**	**AEF % diff.**
A: Female	2.4 ±1.2	0.90 ±0.027	0.88 ±0.058	14.9 ±12.7
B: Male	3.2 ±1.6	0.87 ±0.038	0.87 ±0.056	17.1 ±20.3
A: Young	2.8 ±1.2	0.88 ±0.033	0.89 ±0.048	14.8 ±12.9
B: Old	3.3 ±1.6	0.88 ±0.037	0.87 ±0.072	18.7 ±19.0
A: Low BMI	2.7 ±1.4	0.90 ±0.032	0.88 ±0.071	15.1 ±13.7
B: High BMI	3.8 ±1.8	0.86 ±0.039	0.85 ±0.092	22.6 ±18.5
A: Normal	4.2 ±2.8	0.86 ±0.048	0.83 ±0.121	27.2 ±22.0
B: HCM	4.5 ±2.5	0.84 ±0.044	0.75 ±0.171	39.6 ±24.8

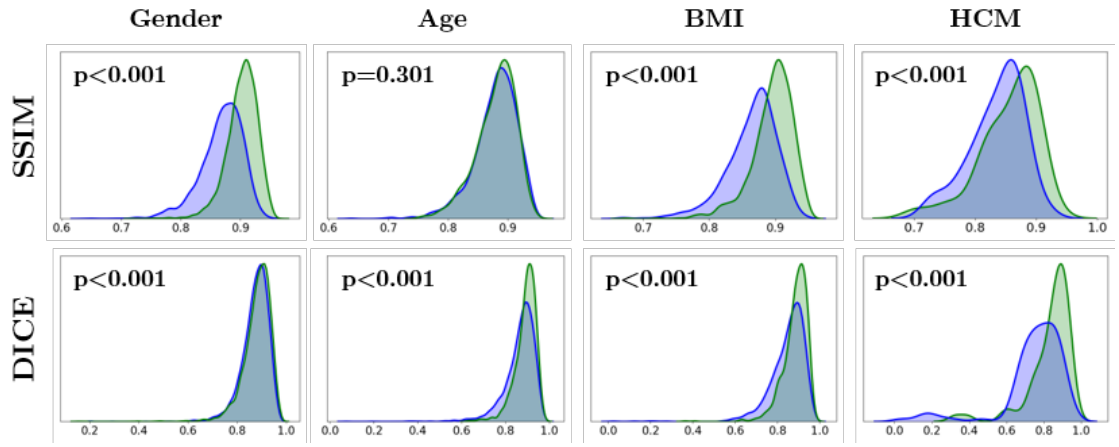


Figure 4.5: Subject Group Histograms; Y-axis= frequency; X-axis= Metric; Group A in green, Group B in blue

calculated p-values based on the KS test. The histograms have been normalised in order to account for the slight class imbalance between HCM and NV cases. Example ES predictions of HCM cases can be be seen in Figure 4.6 and 4.7. Furthermore, the average LV segmentations of ES ground truths and ES predictions, as well as the difference between such, have been computed for all patient groups, including HCM in Figure 4.9. To generate this figure, all test set ES LV ground truth and predicted segmentations were averaged for all patient groups, including HCM. These average ground truth and predicted segmentations were then subtracted from one another to generate the average segmentation difference for each group. Hereby blue areas indicate regions where LV myocardium was found in ES predictions, but not in ES ground truths, whereas yellow parts indicate the opposite. For example, if the prediction for one group consistently underestimated the contraction of the anterior wall of the LV, then the endocardial contour at this position would lie further outward from the LV center in the prediction compared to the ground truth. Hence on the segmentation difference a yellow ring could be seen on the endocardial contour at the anterior wall.

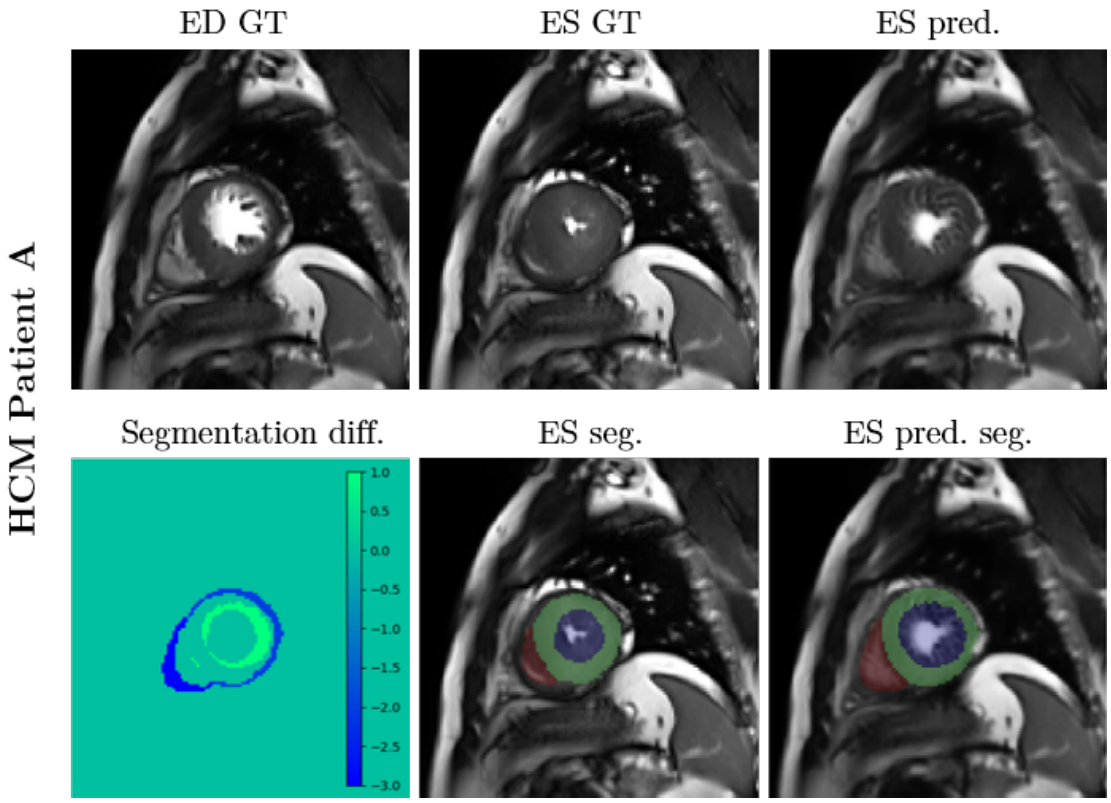


Figure 4.6: cGAN prediction of ES frame for HCM Patient A, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0075, 0.81 & 0.74] respectively.

4.1.7 Discussion

General healthy cardiac motion

The prediction of ES frames, shown in Figure 4.3 and 4.4, can be evaluated qualitatively and quantitatively. Visual inspection shows that the cGAN is capable of predicting ES frames with a high degree of accuracy resulting in remarkably realistic predictions. The cGAN can accurately predict LV myocardial wall movements and ES shapes, showing clear delineations of myocardial contours. RV contraction is also captured. It needs to be emphasised, however, that the cGAN does not explicitly capture motion in the sense of predicting motion vector fields, but rather predicts cardiac shapes as a result of cardiac motion.

Quantitative evaluation confirms the quality of cGAN performance with low MSE scores and SSIM scores close to 1. Yet, since these metrics take into account differences across the entire image, local myocardial deformations might be masked

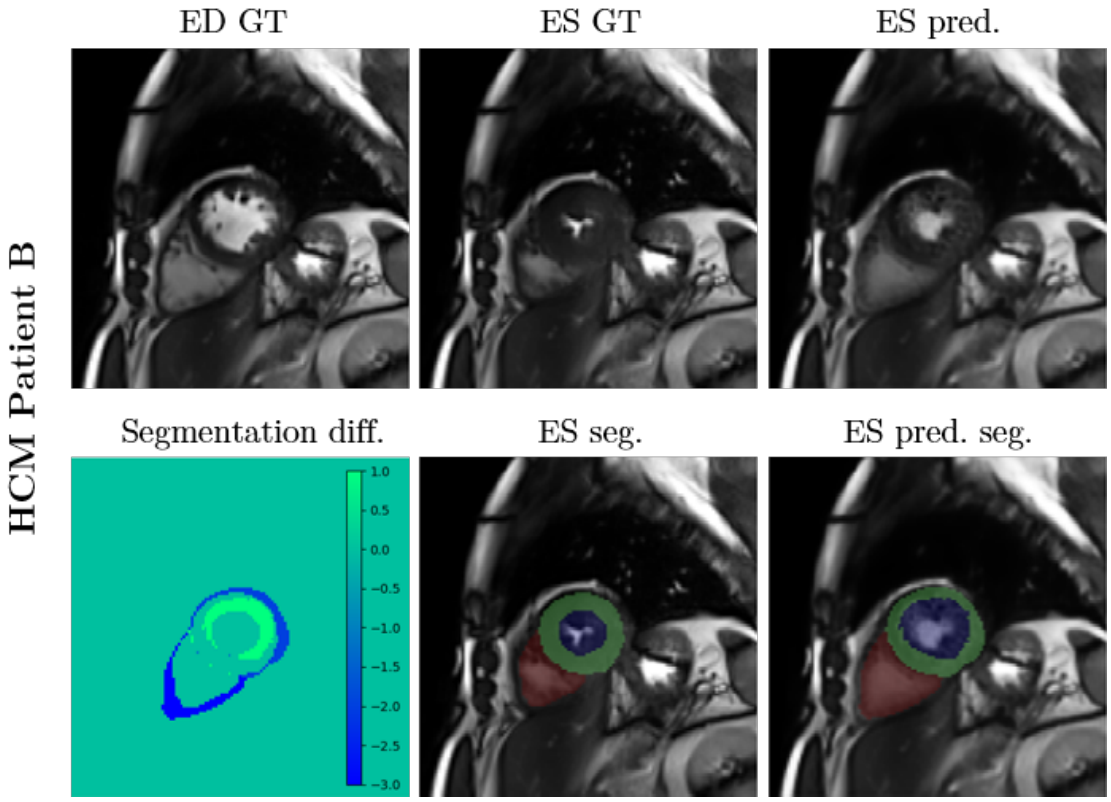


Figure 4.7: cGAN prediction of ES frame for HCM Patient B, including segmentations and difference between ES seg and ES pred.seg.; MSE, SSIM and Dice scores of [0.0032, 0.86 & 0.67] respectively.

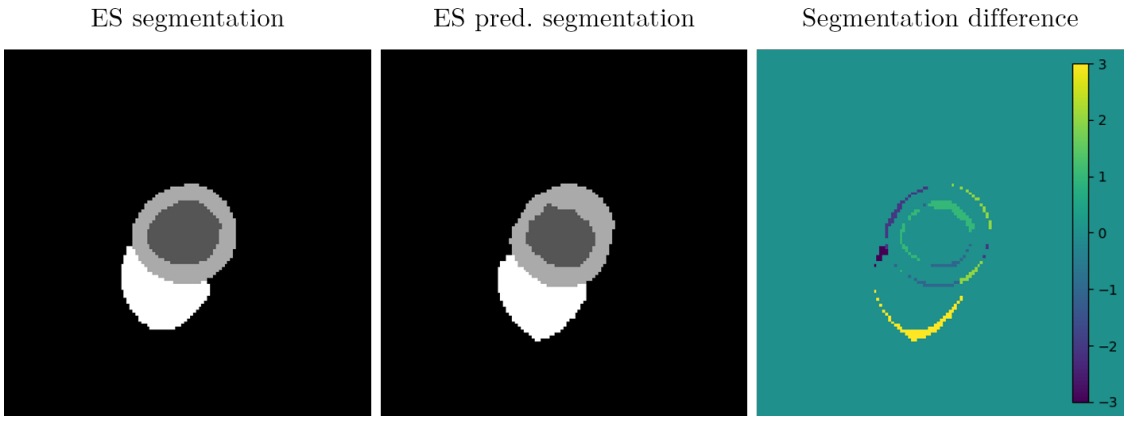


Figure 4.8: Segmentations of ES ground truth and ES prediction and difference between segmentations for high-BMI patient when applied to low-BMI trained network.

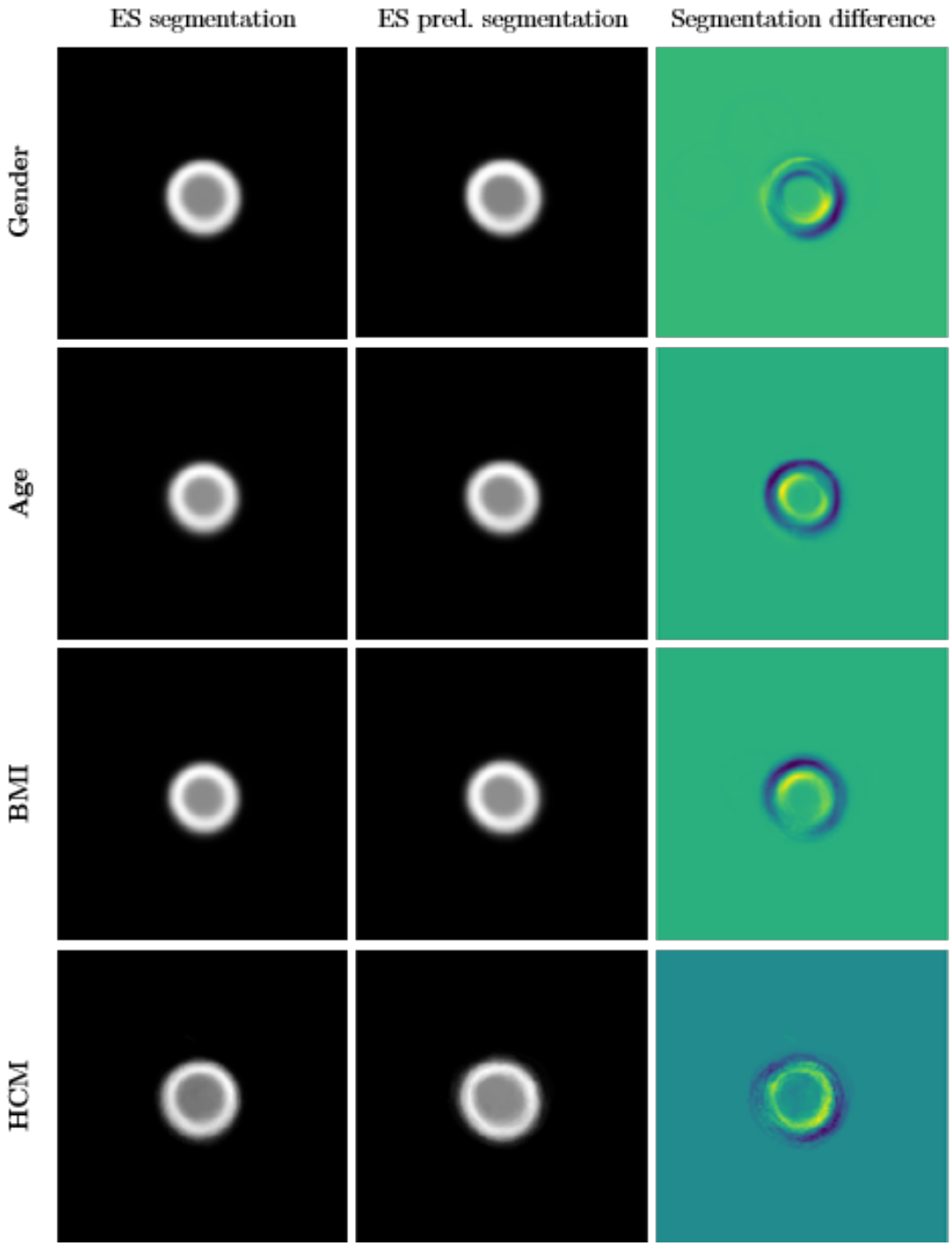


Figure 4.9: Average LV segmentation outputs of ES ground truth and ES predictions for Group B patients when applied to network trained on respective Group A data. The blue part indicates the areas where LV myocardium was found in ES predictions, but not in ES ground truths, whereas the yellow part indicates the opposite.

by larger feature changes located outside the heart. Hence Dice coefficients based on LV cavity segmentations were used to compare LV shapes. The segmentations showed high agreement between the ground truth and predicted ES frames, which confirms that the cGAN has captured the structure-function interactions in a diverse population of healthy volunteers, thereby validating the first hypothesis.

Nevertheless some limitations can be appreciated, particularly on the RV shape which occasionally lacks crisp delineation of its contours. The edges of the LV myocardium also experience occasional blurriness. The intensities of pixels within the myocardium are sometimes not as homogeneous as in the ground truth. Furthermore, some images display difficulties with capturing papillary muscles or valve movements.

Subject group differences

Figure 4.5 and table 4.1, together with their p-values given in the figure, show that when trained on a specific sub-group, the cGAN performs significantly better when tested on separate subjects from that group, as compared to subjects not from that group. Whilst MSE/SSIM/Dice/AEF difference scores are significantly better for trained populations across all tested groups, the level of difference varies between patient group categories. For gender, statistically significant differences can predominately be observed in MSE/SSIM values. This could potentially indicate that image differences other than heart deformation between male and female subjects are responsible for the performance difference. For age groups, the opposite is true. Whilst MSE/SSIM values are more similar, Dice scores differ significantly, indicating significant differences in mechanical function between volunteers of different age groups. In the case of BMI, significant differences can be observed in both Dice and MSE/SSIM scores.

Looking at the differences between the segmentations of the ES ground truth and the ES prediction for a high-BMI patient when applied to a low-BMI trained network in Figure 4.8, an understanding of what this patient's ES image should have looked like if the patient was in fact of low BMI can be derived. The differences in segmentations shows that if the patient were to have a low BMI, the contraction

should have resulted in a more elongated RV, as well as smaller LV cavity. However, the ES ground truth shows a larger LV cavity as well as enlarged LV myocardium.

To gain a more general understanding of patient group specific contractions, and to identify whether this patient specific analysis holds true across the general population, this analysis has been extended over the entire test data population for all subject groups. Results can be seen in Figure 4.9. In the segmentation difference plots, the blue part indicates the areas where LV myocardium was found in ES predictions, but not in ES ground truths, whereas the yellow part indicates the opposite. This means the blue part indicates what the contraction should have looked like if the patient was in fact of the trained group A (female/young/low-BMI), whereas the yellow indicates what the actual group B (male/old/high-BMI) contraction looks like.

In the case of gender, an outward shift of the LV posterior wall in female patients compared to male patients can be seen. For age, there is an almost radial expansion of the LV lumen for younger patients compared to older patients. In low-BMI patients an outwards shift particularly of the LV anterior wall can be observed, creating a larger LV cavity. This method could potentially be expanded to be used in a classification application whereby the ground truth ES image is matched with all potential subject-group specific predictions.

Pathological cardiac motion in HCM patients

Similar to the subject group re-training, it can be seen that the network performs significantly better across all metrics when presented with data from normal volunteers compared to when given HCM patient data. On a qualitative side, as seen in Figure 4.6 and 4.7, it can be noted that predictions and segmentations in HCM cases predominately fail around the LV endocardial contour. The LV myocardial contraction as seen in some ES frames from HCM cases is extreme, with little blood pool left visible. The network often fails to predict that, most likely because it is clearly a pathological mechanism. Furthermore, the network often fails to accurately predict RV shortening in HCM cases, as seen from the

segmentation differences in Figures 4.6 and 4.7. Though it is not just the ES prediction that performs sub-optimally, also the segmentation method struggles. A small number of frames is not even being segmented at all, causing the small low Dice tail in the HCM Dice histogram in Figure 4.5.

While the performance of the network is worse on the NV cases compared to the healthy UK Biobank cases, this can be attributed to the fact that the NV cases originate from a different study. Though steps have been taken to match image appearance, there might be residual differences in the pixel intensity distributions, causing this shift in network performance. Nevertheless, this is irrelevant for the question at hand, as it is equally the case for the HCM cases and it is the comparison of NV to HCM that is of interest. The analysis of average segmentations in Figure 4.9 has also been expanded to the HCM data. Here it can be seen that NV on average had a larger LV cavity, particularly with an outwards shift of the LV posterior wall compared to HCM patients.

When computing the AEF of the ES prediction and the ES ground truth with respect to the ED segmentation, we get mean values of 32.8±13.4 and 53.8±10.5 for the ES prediction and ground truth respectively. In accordance with the qualitative assessment made before, this confirms that the HCM ground truths on average have a higher area ejection fraction than the ES predictions. In other words, when the case is treated as a "healthy" heart and a "healthy" ES prediction is made, the resulting AEF is lower than the actual AEF for that HCM case. This agrees with values in the literature. The same data has been studied in [252] where it was shown that HCM patients have significantly higher ejection fractions than the normal volunteers. Furthermore, the American Heart Association [256] indicates that elevated ejection fractions are linked to HCM. The calculated values for AEF in this case are significantly lower than actual ejection fractions, however, that is most likely because we are considering 2D areas rather than 3D volumes.

4.1.8 Conclusion

The difference in the metrics clearly suggests that the network has learned what "normal deformation" during the cardiac cycle looks like, because when subjected to a different population, where a physiological difference might be expected [257], the ES predictions are not as accurate. Whilst this performance may be expected, it has to be emphasised that both test groups in the second experiment were drawn from the same study and only varied in the test variable (age, gender and BMI respectively) whilst the distribution of the all other variables was consistent across the groups. Hence a difference in performance could potentially not be present and therefore requires experimental evaluation. The positive outcome of this experiment validates the hypothesis that cGANs can be used to learn structure-function interactions unique to specific subject groups.

This is furthermore confirmed when applying it to pathological data where differences due to pathological cardiac function could be detected. As a possible extension, further anatomical and physiological understanding of structure-function interactions and clinical outcomes could be derived from analysing the regions with greatest differences between prediction and ground truth. The technique could potentially be improved by training the cGAN directly on segmented ED-ES image pairs. Advantages of this approach include the focus on the relevant area around the heart and the reduction of ambiguity in fuzzy areas, however, disadvantages are that the segmentations themselves have to be very precise to begin with and that organ texture for example in the form of intensity variations within the myocardium are not taken into account. Particularly in the case of HCM data, a segmentation network trained on HCM and NV cases would outperform the one used in this paper and might lead to more accurate comparisons. The network can also be extended to 3D data which would allow for more complex shape features to be extracted.

Comparable works dealing with ED to ES transformation based on full CMR images using cGANs are unknown to the author, however, similar works exist. These include for example that of Biffi et al. [186, 236] who use a VAE for reconstructing ED and ES segmentation volumes and using the latent space to classify patient

outcomes. Furthermore that of Bello et al. [230] who use a denoising autoencoder to calculate deformation fields along the entire cardiac sequence. Lastly Krebs et al. [225] who use cVAEs that take in the whole cardiac sequence and derive deformation fields for probabilistic motion modelling.

4.2 Metadata Inclusion

4.2.1 Introduction

Physiological differences in cardiac structure and function have been demonstrated to exist between subject groups [257]. In the previous section it was shown that these could be extracted by using cGANs that are trained on specific data subgroups of subjects, therefore learning subject group specific information. This, however, is not a scalable approach, as it would require the training of a new individual network for each additional data subgroup. Therefore the idea was to include subject metadata (such as age, sex, BMI, etc.) directly into the training process and to jointly train the cGAN on image and metadata inputs. This would allow the network to effectively learn distinct sub-networks relating to each metadata group within itself which can be activated during test time when subject data with specific meta information requires modelling. The aim of this section is to investigate the efficacy of such a metadata inclusion in cGANs, particularly as an alternative to training individual networks.

4.2.2 Outline

Firstly, it is of interest to investigate the impact that the inclusion of metadata has on the performance of the cGAN when trained on a mixed group of subjects, i.e. how does the addition of metadata impact ES frame prediction accuracy. To do so, the performance of a variety of cGAN architectures, each with a different method of metadata inclusion, as seen in Figure 4.12, was examined and compared to that of a standard, non-meta cGAN.

Secondly, it was tested whether training a cGAN with meta information could serve as an alternative to training individual networks on sub-populations of subjects. To do so, the best performing meta cGAN architecture, as determined in the first task, was chosen and trained on a mixed population of subjects, together with corresponding metadata. It was then tested with a mixed population, however, incorrect metadata was given to half the subjects, i.e. male subjects were given

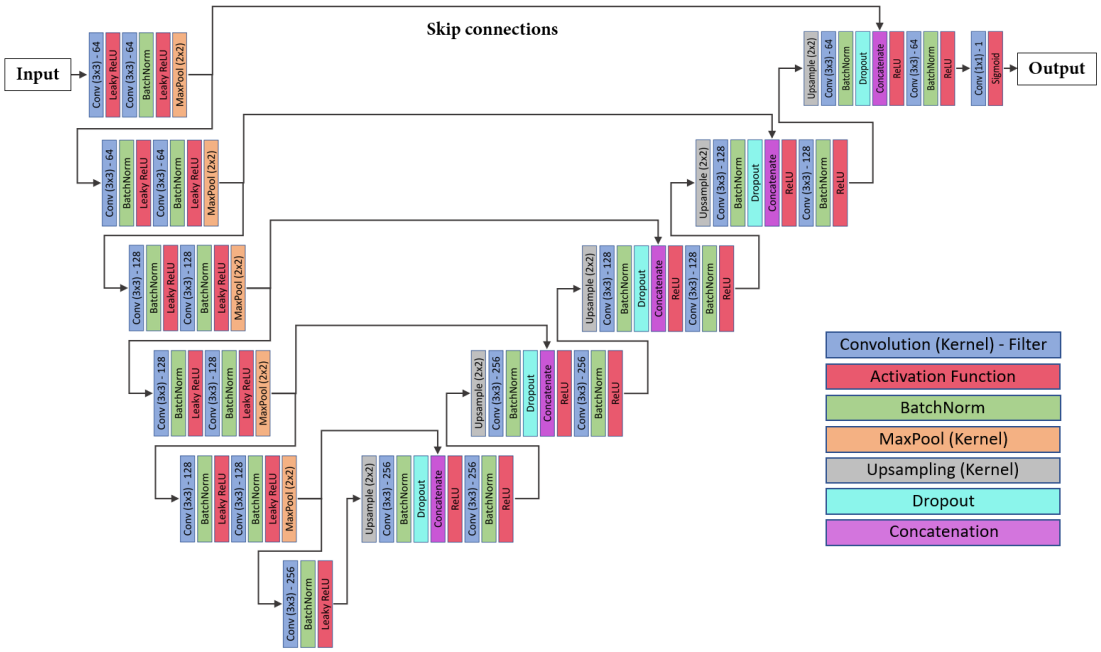


Figure 4.10: Standard cGAN generator architecture.

female subject metadata with the idea that they get treated by the network as female cases. This was to simulate a network that had been trained on only one subgroup of subjects, but then tested on two, as was previously done in section 4.1 where the network was trained on just female cases but tested on both female and male cases.

Lastly, the meta cGAN was jointly re-trained with a wider range of meta-data information, i.e. gender, age and BMI information, and the effect of different metadata input combinations on ES predictions was investigated.

4.2.3 Metadata vs No Metadata

All metadata cGAN architectures were be compared in their performance against the standard cGAN architecture outlined in 4.1. The number of filters and convolutions in the generator and discriminator has, however, been adapted following further development and is outlined in Figures 4.10 and 4.11.

Each network architecture was trained on a total of 10,000 subjects from a mixed population of subjects with roughly equal numbers of female and male subjects and distributions of ages and BMIs that reflect those present in the UK Biobank. For gender information binary metadata was used where "0" represented female

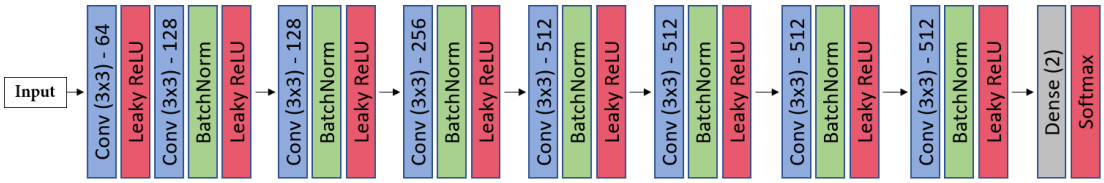


Figure 4.11: Standard cGAN discriminator architecture.

and "1" represented male. Age and BMI information was scaled to a range of $[0, 1]$. Ages in the UK Biobank data ranged from 40 to 69 years, BMI values ranged from 16.3 to 49.1. Each architecture was trained separately with each meta group, i.e. the first time each ED/ES image pair was given additional gender information, the second time age information and lastly BMI information for the last training round. The average performance for all three test runs was also computed. Each network was tested on a set of 2,000 test subjects following similar distributions of metadata to those seen in the training set. MSE, SSIM and Dice scores for cavity and myocardium were computed as test metrics. The results from all test subjects were averaged to give the final results for each network variant.

Metadata cGAN Architectures

There are five different cGAN metadata architectures that were trialed. These include metadata as an additional network input at either, a) the first layer of the generator, or b) the latent layer of the generator, or c) the first and latent layer of the generator, or d) all encoding layers of the generator (including first and latent layer) or e) the first layer of the discriminator. These are henceforth referred to as First Layer, Latent Layer, First+Latent, All Layer and Discriminator First Layer network. For a visualisation of the different architectures, see Figure 4.12.

In a), the First Layer inclusion, the metadata was added as a second input channel to the generator in the form of a 2D image equal to the input image in size, but with every pixel equal to the respective metadata value. This ensures that the metadata inclusion is location independent. For example, the metadata category “gender” was introduced as a 128×128 wide pixel array with all pixel values equal to 0 in the case of a female subject, or 1 in case of a male subject. By

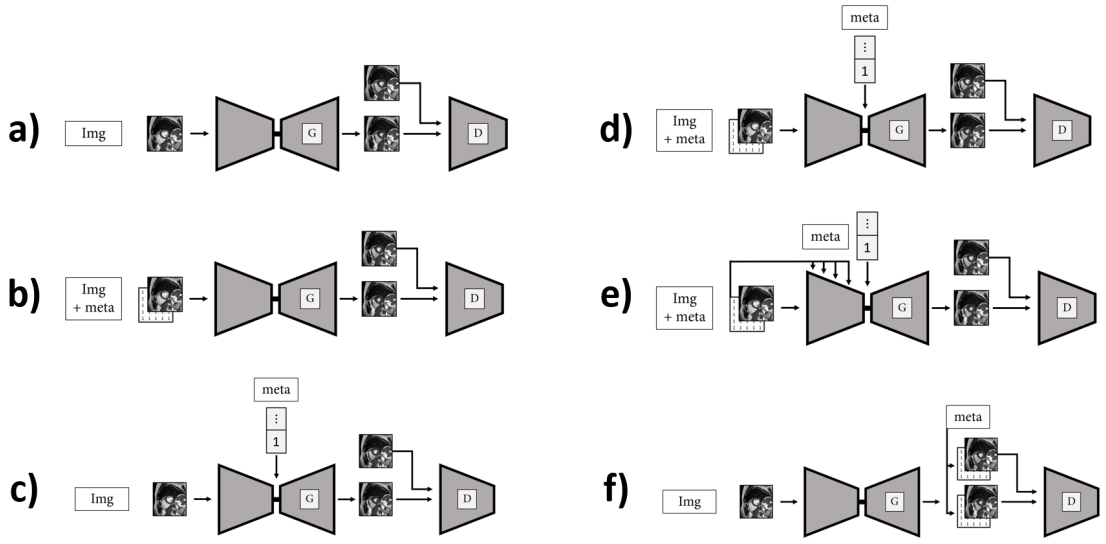


Figure 4.12: cGAN meta architectures: a) Standard cGAN architecture b) First Layer metadata inclusion c) Latent Layer metadata inclusion d) First+Latent Layer metadata inclusion e) All Layer metadata inclusion f) Discriminator First Layer metadata inclusion.

introducing the data at the beginning of the UNET generator structure, it can be ensured that all parts of the network are conditioned on the metadata.

In b), the Latent Layer inclusion, the meta information was concatenated to the latent vector in the generator network. This vector is generated by taking the output of the lowest dimensional layer in the architecture, i.e. the 256 4×4 filters in the bottleneck of the generator, and flattening it. The metadata is then added to this flattened vector as an additional element. After the metadata addition, this vector is then densely connected to the next flat layer in order to achieve the necessary dimensionality for reshaping the vector into a 2D filter before the upsampling process begins. This method ensures that the metadata gets connected to the entirety of the image space and therefore becomes location independent. Including meta information in the latent space seems a suitable method to investigate, as the latent space forms the point in the network at which the learnt features in the image translation process are most condensed and reduced in dimensionality. Metadata inclusion might be most impactful at this position. Issues with this approach are mainly rooted in the fact that the generator takes a UNET structure with skip connections. Therefore the deconvolutional (ascending) branch of the UNET

receives high-frequency inputs from the descending branch which have not actually seen any metadata input and are therefore independent of such. An autoencoder structure without skip connections would avoid this issue, but comes at the expense of reduced performance and is therefore not investigated.

Given the respective advantages of both the First Layer and Latent Layer metadata inclusions, the next step consisted of using both methods in combination in architecture c). Thereby both the more direct addition of the meta input to the latent features and the avoidance of meta-independent skip connections are considered. To further increase the effect of the metadata inclusion, in architecture d) the data was additionally added to each input in the descending (encoding) branch of the UNET following a downsampling step, in a similar fashion as done in the First Layer approach.

As shown in the literature [116], often metadata is not just applied to the generator, but also to the discriminator to facilitate better learning. The reason for this is that if hearts from sub-populations beat differently then providing the discriminator with the corresponding metadata should allow it to be more selective, making the classification easier. Since G trains on the loss of D, this will in turn improve the generation process. In architecture e) the metadata is included in the first layer of the discriminator, similarly to the first layer inclusion in the generator, as an image concatenated the input image.

Results & Discussion

Firstly, the performance of the standard cGAN, i.e. architecture a) as seen in Figure 4.12, needs comparing to that in section 4.1 which was equally trained on a mixed population without metadata. It can be seen that the updated architecture yields significantly better results in ES prediction accuracy. Average MSE and SSIM scores have improved from 0.0026 to 0.0024 and 0.89 to 0.91 respectively. Dice scores have improved from 0.89 to 0.905 for cavity segmentations. Reasons for this improvement are likely linked to the increased network capacity. Furthermore, it has been shown that increasing the minimum dimension of the generator bottleneck

Table 4.2: MSE scores (in 10^{-3}s) for different network architectures when given different meta types as meta input. Note that lower MSE indicate better predictions. Best overall average score and best scores for each architecture across the three groups are given in bold.

MSE scores		***Meta Types***			
		Gender	**Age**	**BMI**	**Avg.**
Architectures	**Standard**	2.428 ±1.246			
	First Layer	2.291 ±1.113	2.314 ±1.044	**2.286** ±1.335	**2.297**
	Latent Layer	**2.299** ±1.200	2.315 ±1.220	2.348 ±1.112	2.321
	First & Latent	**2.314** ±2.068	2.321 ±1.122	2.355 ±1.063	2.330
	All Layers	2.329 ±1.173	**2.324** ±1.262	2.346 ±1.140	2.333
	Discriminator	2.339 ±1.355	**2.262** ±1.295	2.315 ±1.521	2.306

Table 4.3: SSIM scores (in 10^{-1}s) for different network architectures when given different meta types as meta input. Best overall average score and best scores for each architecture across the three groups are given in bold.

SSIM scores		***Meta Types***			
		Gender	**Age**	**BMI**	**Avg.**
Architectures	**Standard**	9.076 ±0.295			
	First Layer	**9.094** ±0.321	9.089 ±0.278	9.080 ±0.283	9.088
	Latent Layer	9.094 ±0.285	**9.104** ±0.271	9.089 ±0.332	9.096
	First & Latent	9.081 ±0.422	**9.092** ±0.273	9.079 ±0.276	9.084
	All Layers	**9.091** ±0.256	9.088 ±0.311	9.080 ±0.270	9.086
	Discriminator	9.094 ±0.312	9.096 ±0.264	**9.105** ±0.335	**9.099**

Table 4.4: Cavity DICE scores for different network architectures when given different meta types as meta input. Best overall average score and best scores for each architecture across the three groups are given in bold.

Cav DICE scores		***Meta Types***			
		Gender	**Age**	**BMI**	**Avg.**
Architectures	**Standard**	**0.905** ±0.082			
	First Layer	**0.907** ±0.054	0.903 ±0.046	0.900 ±0.049	0.903
	Latent Layer	0.901 ±0.045	**0.907** ±0.060	0.897 ±0.044	0.902
	First & Latent	0.905 ±0.052	0.901 ±0.081	**0.906** ±0.049	0.904
	All Layers	0.905 ±0.049	0.901 ±0.047	**0.906** ±0.047	0.904
	Discriminator	0.902 ±0.061	0.903 ±0.049	**0.906** ±0.052	0.904

Table 4.5: Myocardium DICE scores for different network architectures when given different meta types as meta input. Best overall average score and best scores for each architecture across the three groups are given in bold.

Myo DICE scores		*Meta Types*			
		Gender	**Age**	**BMI**	**Avg.**
Architectures	**Standard**	0.855 ±0.094			
	First Layer	**0.863** ±0.078	0.860 ±0.058	0.857 ±0.049	0.860
	Latent Layer	**0.861** ±0.064	**0.861** ±0.073	0.859 ±0.044	0.860
	First & Latent	0.858 ±0.067	**0.860** ±0.090	0.858 ±0.070	0.859
	All Layers	**0.862** ±0.059	0.858 ±0.062	0.861 ±0.047	**0.861**
	Discriminator	0.858 ±0.073	0.858 ±0.054	**0.862** ±0.053	0.860

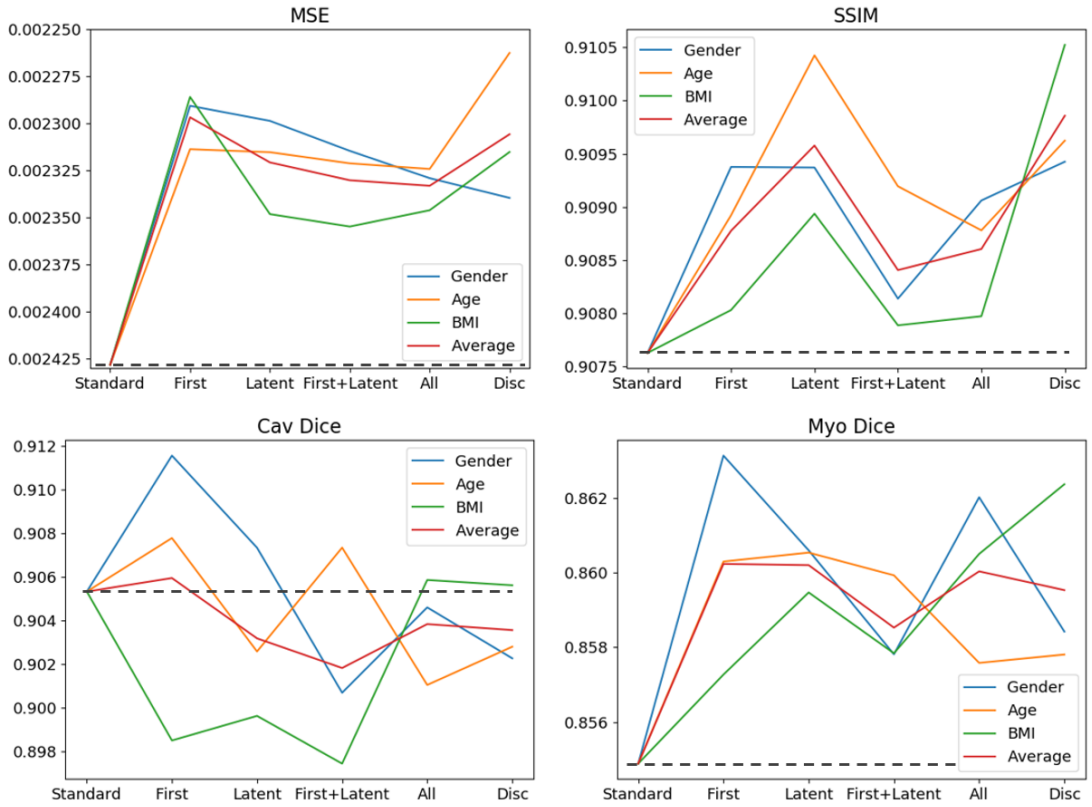


Figure 4.13: Comparision of standard (non-meta) cGAN with different meta cGAN architectures based on MSE, SSIM and DICE scores. The horizontal dashed black line indicates the score of the standard cGAN. Note that the y-axis on the MSE graph has been inverted.

from a 1x1 shape seems to have a positive effect on network performance. This is likely related to the fact that more localisation information can be retained.

Secondly, and being the objective of this section, it can be argued that, based on the data in tables 4.2, 4.3, 4.4 and 4.5 and the corresponding graphs in Figure 4.13, the inclusion of metadata has a positive impact on the prediction accuracy of ES frames in the cGAN. MSE, SSIM and myocardium DICE scores improve with any of the employed cGAN meta architectures compared to the standard, non-meta cGAN results. Cavity DICE scores show improvement for some architecture and metadata combinations, but not for all, with average performance scores only improving for the First Layer architecture. When analysing the results one needs to be careful to consider relative rather than absolute score differences. Particularly for MSE values where absolute differences of 1.4×10^{-4}, as seen between standard and First Layer networks, might seem insignificant, this represents a >6% improvement. Overall, the trend in the results seems to agree with [88] who argued that adding label information almost always improves GAN performance. Between all four metrics, certain trends can be identified. First Layer and Discriminator First Layer metadata inclusion architectures have outperformed other networks on both MSE and SSIM metrics, with the First Layer architecture scoring highest in both DICE metrics as well. Likewise the First+Latent architecture corresponds to the lowest overall performance improvement. Whilst gender and age meta information are comparable in their influence on cGAN performance, BMI meta information has scored lowest in 60% of network metric combinations. In the previous section, having the correct BMI information seemed to be more influential on network performance than having the correct age information, however, those results were firstly achieved using a different network architecture, and secondly relied only included extreme values. Overall, this experiment supports the hypothesis of including metadata in cardiac motion modelling.

4.2.4 Individual vs Joint Network

Introduction

Having established that using metadata networks can generally have a positive impact on ES prediction accuracy, the next step was to investigate how training a single meta network compares to using multiple individual non-meta networks for specific data sub-populations. It is of interest to see if similar results can be achieved, as this would allow for easier scaling of network training applications using data sets with extensive meta information. To set a non-meta baseline, the subject data sub-population training which was previously done in section 4.1 was repeated with the improved cGAN architecture. Hereby, the network was trained on ∼5000 pairs of ED and ES images of just female subjects with ∼1500 pairs of female and male subjects for testing. For time reasons this experiment was only performed using gender sub-populations without conducting separate experiments for age or BMI sub-groups.

As the meta cGAN counterpart, the previously used First Layer network architecture was employed, as it performed best out of all tested meta network variant in the previous experiment. It was re-trained with the same total number of subjects (∼5000), this time, however, stemming from a mixed population rather than from just a female sub-population. The re-training with just ∼5000 subjects was an important step to eliminate the effect of an otherwise significant difference in training data size.

For the testing of the meta cGAN the metadata was initially set to its correct setting, i.e. labelling female subjects as female and male subjects as male. Later, testing was repeated, however, this time with flipped labels, i.e. male data given female labels and vice versa, to simulate the case of testing a network that has previously been pre-trained on a different sub-population. This was thought to be equivalent and comparable to the experiments from section 4.1. There, the network was initially trained on a female sub-population of patients and later given male data for testing. It thus treated the male data as if stemming from a female sub-population, thereby causing the deviation in predictions that were seen in section 4.1. With the meta network the same principle applies. When given an

Table 4.6: MSE, SSIM and DICE scores for two networks. A) Training a non-meta standard cGAN on just female subjects, then testing separately on female and male subjects. B) Training a First Layer meta cGAN on a mixed population of male and female subjects, then testing on a mixed population of male and female subjects separately with correct and incorrect gender meta labels.

Network	Test Set	MSE	SSIM	DICE
a) Non-meta	**Female**	0.00242 ±0.00142	0.911 ±0.025	0.897 ±0.053
	Male	0.00251 ±0.00194	0.897 ±0.053	0.884 ±0.060
b) Meta	**Correct**	0.00245 ±0.00212	0.908 ±0.033	0.892 ±0.061
	Incorrect	0.00252 ±0.00254	0.902 ±0.036	0.881 ±0.073

incorrect or flipped label, i.e. a female label for male data, the meta network's female sub-network is activated, therefore treating the male data as if stemming from a female sub-population, hence mirroring the testing in section 4.1.

Results

Re-training the standard cGAN on just female data and testing on both female and male data has produced average MSE, SSIM and DICE scores which can be found in table 4.6. Equally the MSE, SSIM and DICE results for the meta cGAN when testing with correct and incorrect metadata can be found in table 4.6.

Discussion

Before comparing the results of the non-meta and meta networks to one another, it is important to note that the results of the non-meta network, i.e. trained on just female subjects and tested on both male and female, are significantly better than those previously shown in section 4.1. This improvement in performance is most likely related to the more developed cGAN network architecture.

Now, when comparing the non-meta to the meta network results, the value pairs to examine are those for Non-meta Female and Correct Meta, and Non-meta Male and Incorrect Meta, as these form the equivalent pairs in either network, i.e. either stemming from the same group as the training set, or not stemming from such. For both, the non-meta network results outperform the meta network results in all three metric categories, albeit not by much. The difference between metrics is $< 1.2\%$ in all cases. This shows that when two sets of data are available,

one with a mixed population and corresponding meta information and one with a sub-population, the sub-population trained network will likely perform better, particularly when tested on more data from said sub-population.

In a practical setting though, this small difference is, however, negligible, as the scenario of having two equally sized data sets of mixed and non-mixed data is unlikely to occur, i.e. if there was a data set of just female subjects, it could be added to the mixed data set to create an even larger mixed data set. When comparing the meta network performance to that of the First Layer network in the previous section, it can be seen to be outperformed quite significantly, with a difference of 6.5% in MSE results.

Having examined both the difference in performance between networks trained in this section and those trained previously it can be concluded that network architecture and training data size have a more important impact on prediction accuracy than the choice between meta or non-meta networks. And given that the choice between non-meta and meta networks is often directly associated with data size, meta networks with larger training sets are much more favourable to use to achieve an overall higher prediction accuracy.

4.2.5 Multiple Metadata Inputs

Introduction

To investigate the impact of metadata inclusion on a more qualitative level, the First Layer meta cGAN was re-trained, with gender, age and BMI metadata at the same time. Hereby, similar to the previous application of the First Layer framework, the meta information from all three categories was turned into 2D images and concatenated to the input ED image of the network.

In order to visualise the impact of metadata inputs on ES predictions, the ED test images were passed through the trained network with one of eight possible metadata combinations. These are the first eight binary numbers which cover all combinations of 1 and 0 for three inputs, i.e. [0,0,0], [0,0,1], [0,1,0], [0,1,1], [1,0,0], [1,0,1], [1,1,0] and [1,1,1]. Each input hereby corresponds to a bit of meta information. For gender

the input bit corresponds to the binary input of male or female, however, for age and BMI, where meta values have been scaled to a range of [0,1], the input bits represents meta values at the extreme ends of their distributions. For example, [1,0,1] would represent a male subject of a very young age with a very high BMI.

For each image in the test data set, the image was passed through the network separately with each of the eight metadata combinations. The ES prediction for the [0,0,0] meta input, which represents that of a female subject of a very low age and a very low BMI, was used as the baseline for each subject. The ES prediction for each of the other seven metadata combinations was subtracted from this baseline to generate a difference image. All baseline ES predictions and all difference images were then averaged to give a mean ES prediction for a [0,0,0] meta input and seven mean differences for each of the remaining meta input combinations. These can be seen in Figure 4.14. Figure 4.15 shows a zoomed version of the [0,1,0] difference image to better highlight differences in predictions. Dark areas hereby depict regions which have been predicted as myocardium in the baseline case but not in the other case. Similarly bright areas highlight the reverse.

In order to determine the combination of metadata inputs that differ the most from one another in their ES predictions, the MSE between each ES prediction for a given ED input was calculated and averaged across the whole test set. These were then used to create a confusion matrix as seen in Figure 4.16.

Results & Discussion

In Fig. 4.14 the relative mean differences between meta ES predictions can be seen. While interpretation of these results is not necessarily straightforward, as the differences are complex, some understanding can be derived. Figure 4.15 for example, shows the difference between the prediction of the baseline case, i.e. the [0,0,0] case which represents a young female subject with a low BMI, to the predictions of a [0,1,0] case, i.e. an old female with low BMI. The only difference to the baseline case in terms of meta information is the age of the subjects. Figure 4.15 indicates two ares of difference as pointed to by the yellow and red arrows. The

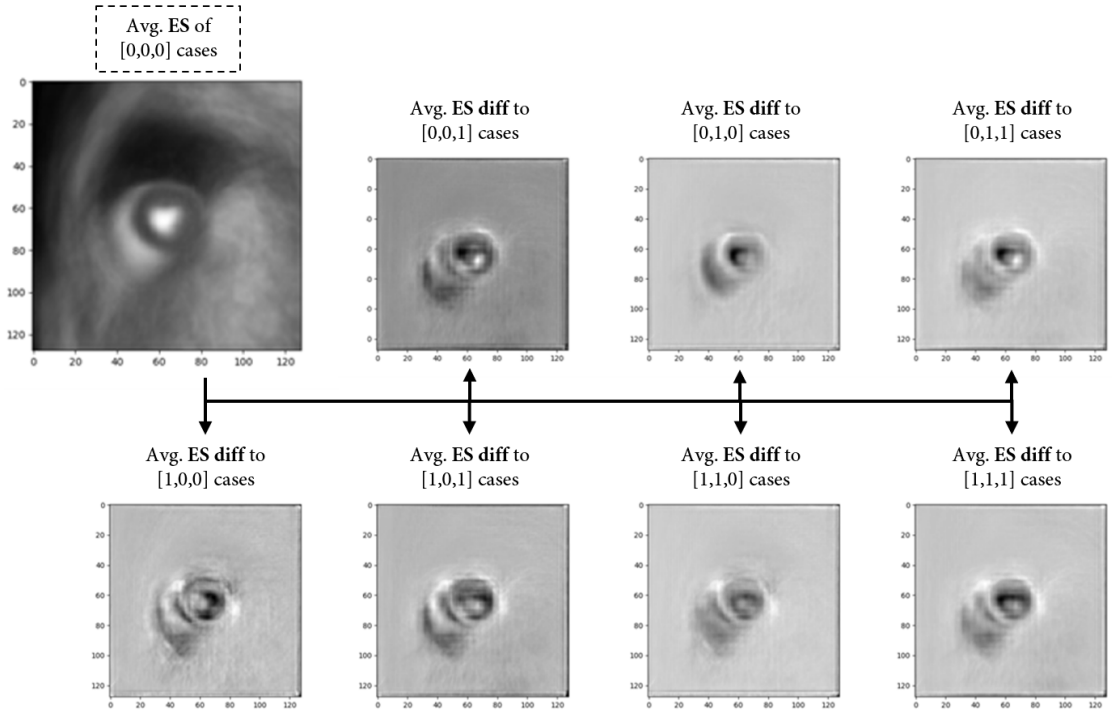


Figure 4.14: Top left image: Average ES output for ED input cases given additional meta information of [0,0,0], where square bracket values indicate [gender, age, BMI]. All other images: Average difference between top left ES outputs with given [0,0,0] meta combination and ES outputs given the respective meta combination indicated. Dark areas indicate myocardium label predictions for [0,0,0] baseline case, yet cavity label predictions for the respective other case. White areas indicate the opposite.

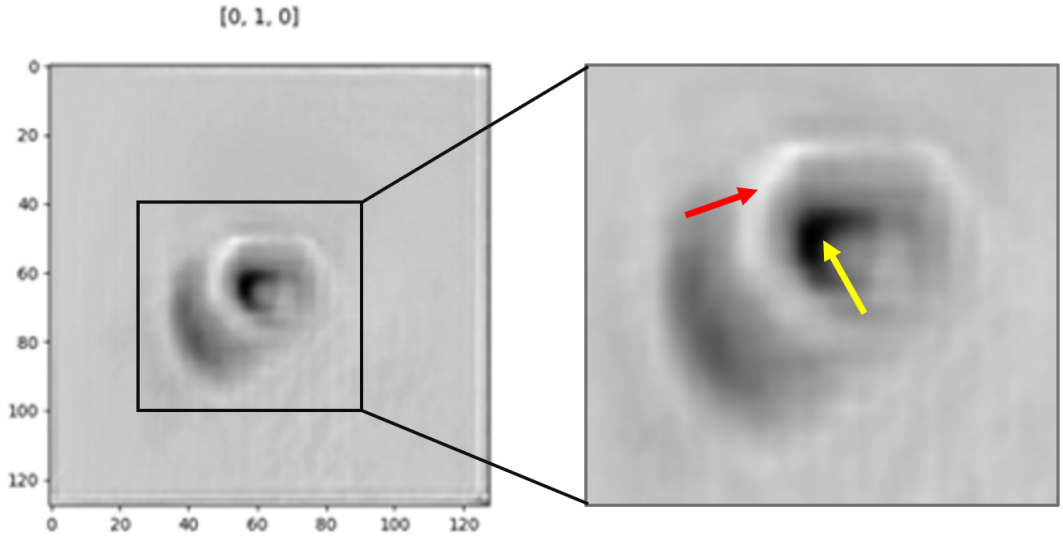


Figure 4.15: Zoomed image of [0,1,0] case from Figure 4.14 indicating areas of over- and under-prediction of myocardium between the [0,0,0] and [0,1,0] meta ES predictions. The red arrow indicates an area that over-prediction between the [0,0,0] and [0,1,0] cases. Respectively the yellow arrow indicates an area of under-prediction.

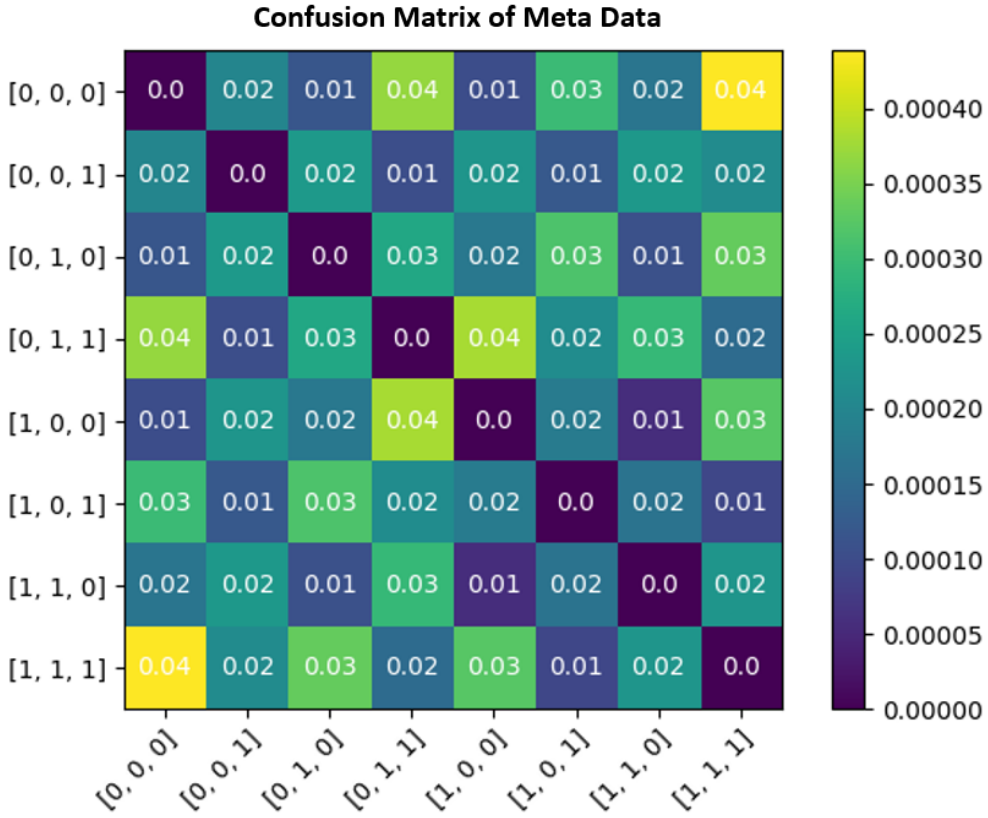


Figure 4.16: Confusion matrix showing array of MSE differences between mean meta outputs in 4.14. All values are in 10^2. Higher values indicate larger difference between given combination of meta outputs.

dark area, as indicated by the yellow area in the epicardium, shows a region where the baseline case predicted myocardium labels, but the [0,1,0] case predicted cavity labels. Equally, the white pixels on the anterior and anteroseptal outer wall of the myocardium, as pointed to by the red arrow, show a region where the baseline case predicted RV labels, but the [0,1,0] case predicted myocardium labels.

Overall this means that on average low BMI female subjects of a higher age seem to have both a slightly LV wall thickness as well as cavities in the ES state compared to low BMI female subjects of a lower age. This agrees to a large degree with the understanding that myocardial contractions alter over time, increased LV wall thickness and reduced ejection fractions [258, 259]. Yet, it needs to be considered that physiological changes and cardiac remodelling are complex processes and inherently three dimensional, hence inferring too much about global function from a mid-axial SAX slice needs to be treated with caution. Nonetheless, the difference

images in Figure 4.14, do allow for some degree of physiological understanding of the differences in contraction between subject sub-groups.

The confusion matrix in Figure 4.16 shows the mean MSE differences between meta ES predictions for different meta combinations. The biggest difference can be found between the [0,0,0] and [1,1,1] cases which represent female, young and low BMI subjects and male, old and high BMI subjects respectively. This is a reasonable outcome, as both cases contain the opposite meta information for all three categories. Next biggest differences can be observed for male, young and low BMI subjects when compared to female, old and high BMI subjects and female, young and low BMI subjects when compared to female, old and high BMI subjects. These two cases seem to indicate that large changes in age and BMI have potentially more significant impacts on the ES prediction than gender. Overall it can be shown that meta information does influence cardiac function and helps shape a more accurate, personal prediction.

4.3 Conclusion

The work in this chapter has shown that cardiac function can be modelled by using a frame-to-frame prediction method based on conditional generative adversarial networks. Hereby cardiac contractions are modelled by conditioning ES frame predictions on structural ED input data. The applied cGAN networks have proven that high levels of prediction accuracy for general mixed populations of clinical subjects can be achieved, with clinical metrics such as ejection fractions found to be comparable to those present in literature.

Applying cGANs to sub-populations of subject data has shown that population specific functional information not only exists, but also that it can be captured by such networks. Population specific function can be used to highlight differences between sub-populations of subjects, as well as between healthy and pathological hearts, such as those suffering from HCM. These functional differences can furthermore be extracted and visualised to provide physiological understanding.

It has been shown that for the 2D image to image based approach, subject metadata inclusion into the frame prediction process yields an improvement in prediction accuracy, based on MSE, SSIM and DICE metrics. Hereby a First Layer approach, whereby the metadata gets fed into the first layer of either the generator or the discriminator seems to achieve the best improvement in performance. It has also been shown that, given data sets of equal size, training a network on a sub-population of data outperforms an equivalent network trained on a mixed population including its corresponding metadata. The increase in performance was, however, marginal. More important seems to be the choice of overall network architecture, as well as the total size of the training data set. Similar to the approach of training on sub-population data in section 4.1, the inclusion of subject metadata allows for the modelling of sub-population specific cardiac function and the visualisation thereof.

Metadata inclusion is an important step in enabling more subject specific cardiac modelling in a scaleable way. However, the method of inclusion seems to matter greatly and is likely to not have been optimised yet in this work. Braided blocks, first published in [260], appear to offer a promising method of achieving better

results in this regard. They work by embedding metadata into image space through a combination of scaling and non-linear activation layers. As a result, the effect of meta inputs on the image space can be maximised.

Additional methods of improving the meta-specific frame prediction could involve using an additional metadata related loss function in the cGAN. Hereby the network would aim to predict the metadata as an additional output in the discriminator which would force the network to place more emphasis on the meta information during the training process. As a result, it will likely extract features that are more discriminant between meta groups during training.

A further extension to this work could consist of including additional metadata into the modelling process. This can include data both non-pathological in nature, such as patient ethnicity, as well as pathological, such as specific cardiovascular disease diagnoses, as classified according to the ICB-10 standard in the UK Biobank. Assuming the First Layer meta network is being used, the addition of new metadata could be performed in two ways. Firstly, the entire network could be re-trained from scratch with all metadata inputs, including the additional ones. Or secondly, the trained network could be fine-tuned with the addition of the extra metadata inputs. Whilst pre-training of networks is generally advantageous and accelerates convergence, it might be that training from scratch could allow the metadata influences to be more equally weighted in directing the network's function. However, this is difficult to predict without further investigation.

With an increasing number of metadata points, the chance that some of the data points are missing for some patients also rises. To account for those gaps in the data and allow the network to function robustly metadata imputation is required. This is the process of estimating missing data points and has been achieved through various methods such as denoising autoencoders and GANs. Yoon et al. [261] have further generalized the GAN setup to allow for imputation even when complete data is unavailable. An adaption of this process into the cGAN structure used in this chapter could be an interesting step.

Section A in the appendix which outlines work done by Beetz, Ossenberg-Engels, et. al. covers the application of how Point Cloud Networks (PCNs) can be used to replicate the function modelling process in a memory efficient 3D way. Additional extensions to this work could include more extensive modelling of cardiac dilation, as already outlined in the PCN section in the appendix. This is of particular interest with regards to pathologies like HCM which particularly affect the dilation process. A further extension to the full cardiac sequence is then the logical next step.

The key contributions of this chapter are:

- Modelling of cardiac contraction using cGAN-based frame-to-frame prediction method with high degree of accuracy as shown by clinical metrics.
- Extraction of sub-population specific cardiac function information with associated visualisation for physiological understanding of sub-population differences in cardiac contraction.
- Extension of sub-population modelling to pathological HCM data.
- Comparison and investigation of different methods of metadata inclusion into cGANs for improved function prediction.
- Comparison of metadata inclusion to training of sub-population data for improved sub-population specific function modelling and scaling of training applications.
- Analysis of effects of including multiple metadata points on ES prediction accuracy and the physiological understanding of different metadata groups in a population.

5

From 2D to 3D: 2-Ch Long-Axis Segmentation

Contents

5.1 Introduction 168
5.1.1 Disclaimer 168
5.1.2 3D networks in CMR 168
5.1.3 Moving from 2D to 3D 169
5.2 Data 171
5.2.1 Choice of Data 171
5.2.2 Segmentation Process 171
5.2.3 Additional Data 174
5.2.4 Pre-processing 175
5.2.5 Data Augmentation 175
5.3 Methods 178
5.3.1 Metrics 178
5.3.2 Network Architectures 178
5.4 Segmentation Network Comparison 179
5.4.1 Results 180
5.4.2 Discussion 180
5.5 Segmentation UNET Optimisation 181
5.5.1 Introduction 181
5.5.2 Hyperparameter Optimisation 182
5.5.3 Results 182
5.5.4 Discussion 183
5.6 Addressing Myocardial Breakage 188
5.6.1 Introduction 188
5.6.2 Methods 189
5.6.3 Results 190

5.6.4 Discussion 190
5.7 Conclusion 193

5.1 Introduction

5.1.1 Disclaimer

While the work in this chapter forms a component of a wider publication by Beetz, M., Banerjee, A., **Ossenberg-Engels, J.**, Grau, V. titled *"Multi-class point cloud completion networks with cross-domain transfer for cardiac surface reconstruction from cine magnetic resonance images"* which is currently being submitted to Medical Image Analysis and on which the thesis author is third author, the work in this chapter was entirely completed by the author of the thesis.

5.1.2 3D networks in CMR

Training 3D DL networks has long been a challenge in the medical image analysis space, primarily due to the limited availability of computational power and 3D data sets of sufficient size, particularly annotated ones. Shifting networks from 2D to 3D is usually accompanied by a significant increase in the number of learnt parameters, often requiring substantial upgrades in GPU power to handle computations. Such upgrades used to come at a considerable financial cost, an investment that would not always be made worthwhile by what sometimes only led to marginal improvements in network performance. Furthermore, any potential improvements were frequently outweighed by the performance deterioration caused by the smaller size of 3D training data sets. Given the combination of larger parameter numbers and smaller training data sets, 3D networks were also more prone to overfitting. In order to combat this effect, alternative networks, including 2.5D approaches where stacks of 3D slices are used, including multi-view and multi-slice networks [262], have been explored.

However, with more capable GPUs and larger medical data sets, such as the UK Biobank, becoming more widely available these issues have become less relevant in recent years and applications of 3D networks are starting to become more common in the field. Still, the nature of the underlying application and the type of data that is available (i.e. 2D or 3D) seem to determine the choice of network in most

cases. It has been shown that in many cases when the underlying data is inherently 2D, 3D networks tend to be outperformed by 2D networks [263]. Yet, for data sets that have been acquired in 3D, such as CT or 3D MRI, 3D networks seem to be performing best [264, 265].

In general, 3D data offers a lot more information than 2D. Specifically, in the case of CMR, 3D CMR data has the advantage of providing a consistent 3D representation at high resolution, thereby minimising the problem of motion that is common in 2D CMR. This ensures a more accurate capture of 3D cardiac shape. Furthermore, the resolution along the z-axis is higher in 3D compared to 2D where gaps between SAX slices tend to be larger than their in-slice resolution. This allows for higher resolution voxel neighbourhood information. Multiple aspects of cardiac structure and function can only be fully understood when observed in 3D, hence capturing and analysing it in 3D offers a more holistic approach to cardiac modelling. Lastly, common 2D pre-processing steps such as correcting slice misalignments and inter-slice registrations, both processes which can add additional uncertainties, are reduced when using data that has been captured in 3D.

It is therefore of interest to use 3D CMR data in the analysis of cardiac structure and function.

5.1.3 Moving from 2D to 3D

In the case where 3D data is not available, it first needs to be generated from 2D data. To do so, a combination of well established processes involving the co-registration of slices, interpolations and augmentations of pre-existing shape meshes exists [266]. However, these processes are often computationally expensive at run time and inefficient in their application to larger data sets. Additionally, slice misalignments, missing slices and other artefacts can heavily affect the quality of the 3D reconstruction.

To accelerate and improve this process a new pipeline using cGANs is envisioned which utilizes 2D segmentations at various orientations (SAX, 2CH, 3CH, 4CH LAX) as its inputs and maps them to a full 3D volume segmentation at inference

time. The more of these orientations that are available for each volume, the more accurate the reconstruction will be. The envisioned network is targeted primarily to work on UK Biobank data. While segmentation data exists for short-axis and 4ch long-axis views, no segmentation data is currently available for 2ch views, nor is a trained segmentation network to produce such. Therefore the first step of this work is to design and train such a network.

5.2 Data

To train a segmentation network, segmentation ground truth training data needs to be available. As no ground truth segmentations of 2ch UK Biobank data were available to the author at the time of this work, new ground truth segmentations had to be manually created.

5.2.1 Choice of Data

For the generation of ground truth segmentations, 2ch long-axis frames needed to be chosen. The choice of which frames to use for this purpose was driven by several factors. Firstly, in order to maximise the robustness and generalisation of the segmentation network, maximum variance in the data was strived for. Therefore each frame was chosen from a different subject in the UK Biobank. Secondly, whilst in the short-term the 2D to 3D network would focus on ED and ES frames, in the long-term the aim was to cover the entire cardiac sequence. With this in mind, the chosen frames were equally distributed across the entire cardiac sequence.

A total of 200 2ch LAX frames were chosen to be segmented from the UK Biobank. This number formed a compromise between time constraints stemming from the segmentation process itself and what was deemed to be a sufficiently large training set based on previous experience and relevant literature. The frames were taken from 200 separate subjects, covering the entire 50 frame sequence four times over, i.e. frame #1 was taken from subject #1 at sequence position #1, frame #2 was taken from subject #2 at sequence position #2, etc., and frame #51 was taken from subject #51 at sequence position #1 and so on.

5.2.2 Segmentation Process

Dr Iulia Popescu, a Postdoctoral Research Fellow at the Radcliffe Department of Medicine in Oxford and an expert in CMR image segmentation was consulted for her advice on the segmentation process. In agreement with her, the epi- and endocardial contours were labelled, as well as the basal contour along the mitral valve, as seen in

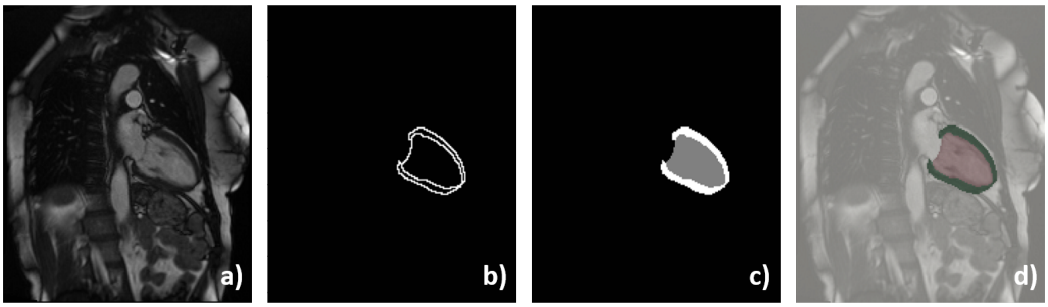


Figure 5.1: a) Example of long-axis 2ch scan from the UK Biobank. b) Corresponding manually segmented endo- and epicardial contours, as well as basal contours. c) Full pixel labels for LV ventricle cavity (in gray) and myocardium (in white). d) Overlay of segmentation labels onto image.

Figure 5.1, allowing for an easy computation of labels for the cavity, myocardium and background. The open source tool ImageJ was used for the segmentation process.

Segmentations in the clinic are often performed using the cvi42 software package whereby fine contour lines are drawn on the segmentation image from which pixel labels are interpolated based on a proprietary interpolation algorithm. In the case of the ImageJ segmentations, a pixel-based segmentation method was followed whereby pixel labels were assigned directly and without the need for interpolation. There are several advantages to both strategies:

A pixel-wise segmentation can introduce certain operator biases that an interpolated contour segmentation does not. For example, if a segmentation contour lies at a non-zero, non-orthogonal angle to the pixel grid, i.e. it runs diagonally across the image, then a single pixel-wide “straight” line is de facto impossible to draw. Instead an indirect line including numerous right-angled turns will exist, due to the pixelation of the image. Whilst the interpolated contour segmentation method would calculate the location of such kink pixels automatically, thereby freeing the operator from having to actively make biased decisions, the pixel-wise segmentation method does not do that. Hence, based on the biases of the operator, certain features may get systematically included or excluded, creating deviations in the segmentation results compared to using an interpolated contour segmentation method.

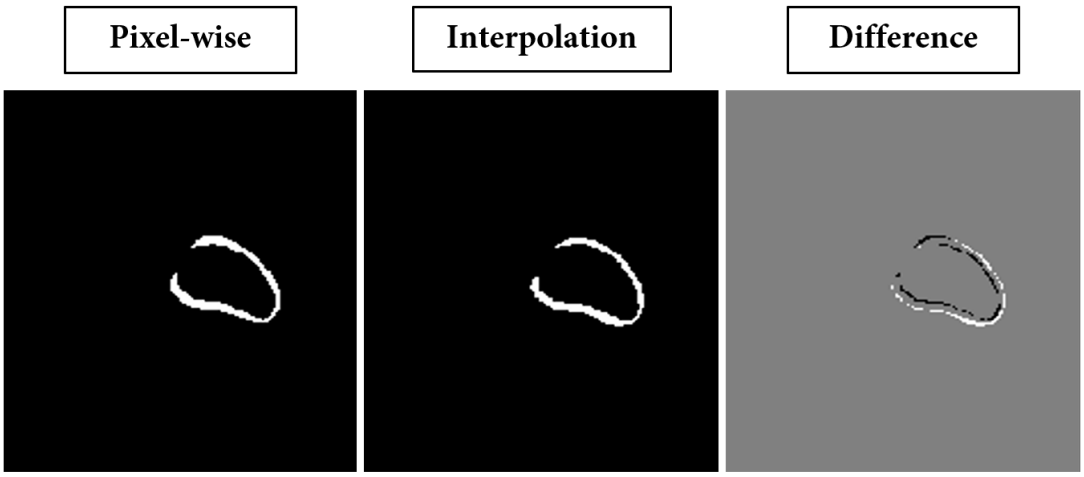


Figure 5.2: Example of the systematic difference that can occur between pixel-wise and interpolation based segmentation methods. The interpolation method systematically labels myocardium pixels further outwards than the pixel-wise method.

The interpolating contour method, however, comes with its own issues. First of all, following up from the previously mentioned point, not having direct control over the assignment of pixel labels can also be a hindrance. Furthermore, depending on the underlying interpolation algorithm and how it decides to assign labels to pixels that are crossed by a drawn line, i.e. only partially included in a specific label region, the contours of the myocardium can inadvertently end up being shifted inwards or outwards, resulting in a systematic over- or underestimation of the labelled regions. An example can be seen in Figure 5.2.

A difficulty for both types of segmentation is the issue of how to deal with papillary muscles, particularly during ES, as these are situated right next to the endocardium and share the same pixel intensities, making them effectively merge with the myocardium into a single structure in the image. This can often become a problem when just looking at the standalone ES frames without having seen prior frames that illustrate the thickening of the myocardium. To identify and avoid this issue, it is usually best to examine the myocardial motion along the cardiac sequence, as this will highlight the papillary muscles during ED and display a merger more clearly. This way papillary muscles can be identified and separated from the myocardial wall despite matching pixel intensities. All other

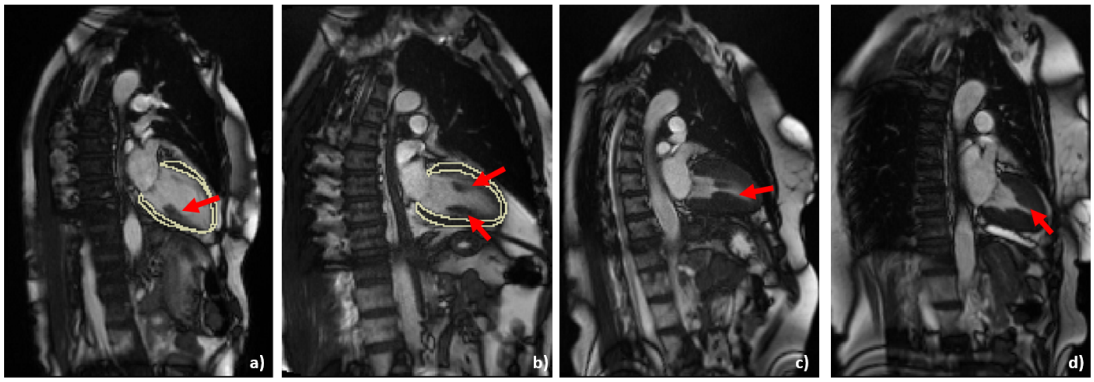


Figure 5.3: Examples of papillary muscles as seen on 2ch long-axis scans. Papillary muscles are depicted as shaded regions in the LV cavity, indicated by the red arrows in a) and b). Due to their similarity in image intensity, they easily merge with the myocardium, particularly in frames close to ES, as seen in c) and d), and care needs as to be taken as to not include them in a myocardium segmentation. Using the whole image sequence to observe the motion of the myocardium hereby usually helps.

non-myocardium components can thereby equally be excluded. This technique is commonly implemented for ground-truth segmentations in the clinic and was used in the segmentation application in this chapter too. An example of this problem is highlighted in Figure 5.3.

The data set used in this work was segmented using a pixel-based approach. Further examples of such segmentations can be found in Figure 5.4.

5.2.3 Additional Data

A second data set of 2ch LAX frames and their ground truth segmentations, comprising 200 ED and ES frames, was supplied by Dr Elena Lukaschuk. These were produced using the cvi42 system with the interpolated contour method. When comparing the two data sets, it became obvious that the interpolation algorithm in Dr Lukaschuk's data had the effect of introducing a systematic difference between the two sets of segmentations. Due to the interpolation process, these segmentations tended to move the myocardium further outwards from the LV centreline, thereby overestimating the LV cavity. An example can be seen in Figure 5.2. This data set was used for additional validation purposes later in the process.

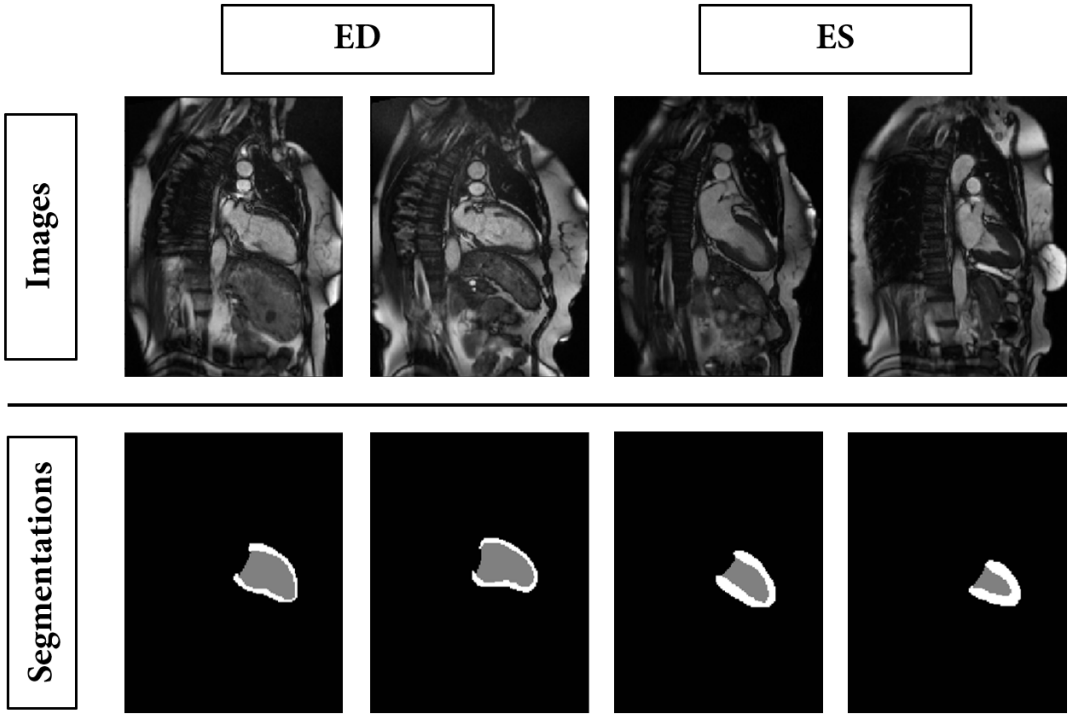


Figure 5.4: Examples for ED and ES long-axis 2ch segmentations from the UK Biobank using a pixel-wise segmentation method. LV cavity pixels are labelled in gray and myocardium pixels in white.

5.2.4 Pre-processing

To improve network performance and reduce computational load, training images and corresponding segmentations were cropped to a size of 128 x 128 pixels, centered on the LV cavity. Instead of using the centre of mass of each image's cavity individually for centering that particular image, the average centre of mass of the LV cavity across the entire image population was computed and all images and segmentations were translated accordingly. This was done to avoid generalisation problems. Centering was followed by an intensity re-scaling of the images, whereby the top 1% and bottom 1% of pixel intensities were squashed to a value of 1 and 0 respectively, with the rest of the pixel values being scaled linearly in between those limits.

5.2.5 Data Augmentation

In order to increase the training set size, training images and segmentations were augmented on the fly with random combinations of rotations, translations and

scalings whilst maintaining the 128 x 128 image dimension. The limits for these transformations were: ± 30 degrees rotational limit, ± 10 pixels translational limit in both x and y and 20% scaling limit either way (zoom in or out). Examples of such augmentations can be found in Figure 5.5. Other forms of augmentation, such as changes in image intensity could also have been applied to further increase data variability. However, the training and testing data sets did not display a high degree of variability in image intensity, due to the fact that the UK Biobank data generally has a high degree of homogeneity. Therefore it was deemed unsuitable to introduce intensity variation as an additional form of augmentation. For applications where data is obtained from systems of multiple vendors, where a higher degree of intensity variation can be expected, this form of augmentation would be more suitable.

Out of the 200 total images, 150 were used for training with 25 for validation and 25 for testing. A total of 10 augmentations per original training image was used leading to a total training set size of 1500 image and segmentation pairs. In order to improve network performance and reduce the impact of small validation and test batches on the robustness of results, three rotations of cross validation were run for each experiment from which results were averaged.

As demonstrated in chapter 3, GANs can be employed to generate realistic data samples, thereby supplementing pre-existing data sets. This can be done in addition to or instead of traditional augmentation techniques. If trained well, GANs are thought to have the advantage of being able to introduce complex variability into a data set, something that rigid augmentations might struggle to achieve [267]. The practical benefit of using GAN-generated augmentations has been successfully demonstrated by Frid-Adar et al. [267] who use them to improve their CNN-based liver lesion classification method. However, [268] have shown that similar levels of performance have been achieved using traditional augmentation techniques compared to using GAN augmentations. For extremely small data sets it might actually be preferential to use traditional techniques, as GANs often struggle to reach sufficiently good levels of performance with small training sets. Lastly, using

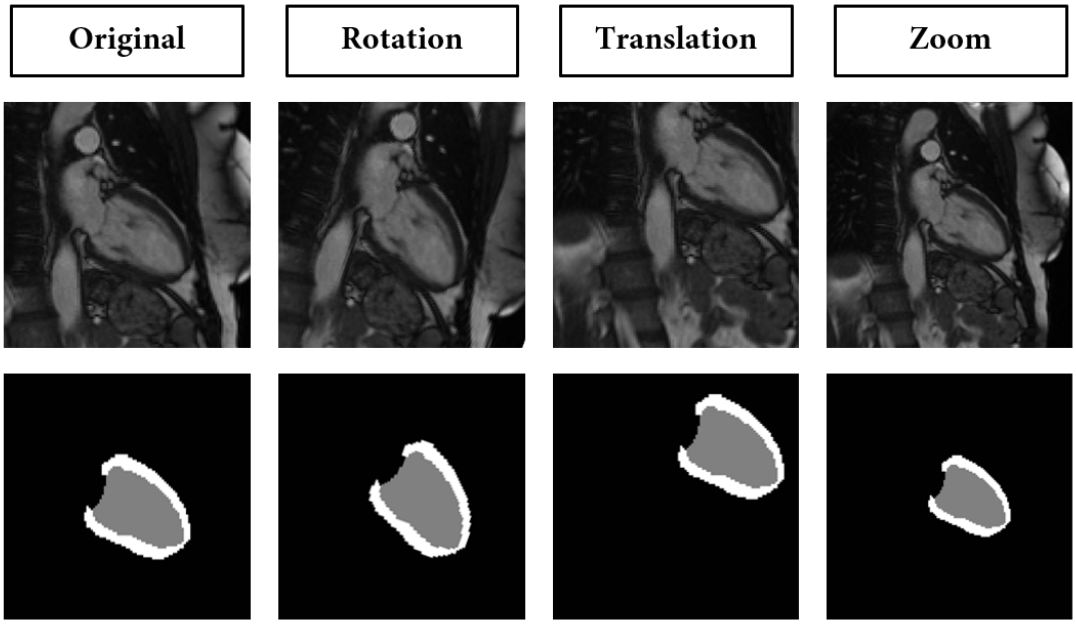


Figure 5.5: Examples of augmentation to expand training data set size, including rotation, translation and zoom.

GANs for augmentations in segmentation tasks means that usually no corresponding ground truth data is available which makes them inadequate for the purpose.

5.3 Methods

5.3.1 Metrics

To design an optimum LAX 2ch segmentation network, several different architectures and hyperparameters were investigated. Their performance was evaluated based on the Dice coefficients of the LV cavity, LV myocardium and combined LV cavity and myocardium. Furthermore, especially in ED segmentations, and observed in the bottom left image in Figure 5.4, the apical myocardium thickness can often reduce to not much more than a single pixel's width, sometimes causing the myocardium segmentation to break into two or more disconnected parts at the apex. This is unwanted, because it makes downstream calculations of clinical metrics more difficult and also because it is anatomically incorrect. The fraction of test cases that present such a myocardial breakage served as another performance metric. A myocardium segmentation was deemed to be "fractured" if it consisted of more than one connected component. Hereby two diagonally adjacent pixels counted as connected, i.e. non-fractured, according to the 4-connectivity rule.

5.3.2 Network Architectures

The first network architecture that was investigated is Bai et al.'s segmentation FCN [204] used for short-axis and long-axis UK Biobank data segmentation and depicted in Figure 5.6. Hereby, multiple layers of convolution-batchnorm-ReLU units followed by a maxpool layer are stacked in sequence. After each layer, the output gets propagated to the next layer, whilst also being upsampled to the original image dimensions using transposed convolutions and concatenated with the other features before undergoing a final convolution. Since this architecture has already been applied to UK Biobank data, it served as a benchmark for segmentation quality evaluation.

To evaluate its suitability it was compared to two other architectures: a UNET-style implementation by Xu et al. outlined in [269] and a second proprietary UNET

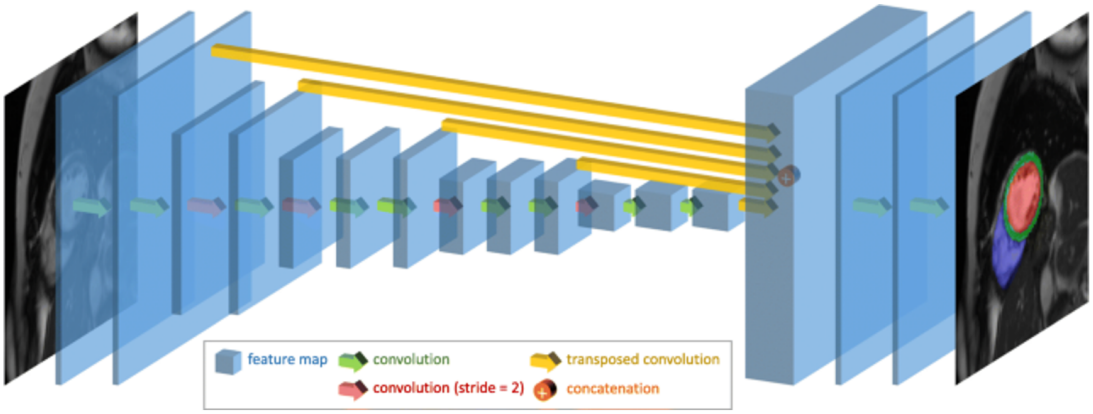


Figure 5.6: Fully Convolutional Network developed by Bai et al. [204] for the segmentation of short-axis and long-axis CMR data from the UK Biobank. Figure taken from [204]. Successive layers of convolutions, batchnorm layers, ReLU activations and maxpooling layers are applied from which outputs serve both as the input to the next downstream layer, as well as being upsampled and concatenated to form the final layer.

implementation which followed the original UNET hyperparameters [66]. The aim was to identify an optimum segmentation architecture for the underlying data.

All three architectures took pre-processed 2ch LAX images as their input and generated three class label probabilities for each pixel as their output. In Bai et al.'s network and in the UNET these classes corresponded to the cavity, myocardium and background, whereas in Xu et al.'s network these corresponded to the cavity, epicardium and background instead. A further post-processing step was therefore required to attain the probabilities for the cavity and myocardium labels from this. The final label for each pixel was assigned to the class with the highest probability for that pixel.

5.4 Segmentation Network Comparison

As previously mentioned, each network was trained and evaluated on three different sets of training and validation data from which average performance results were computed. Once the best of the three given architectures was determined, the winning network underwent a series of hyperparameter evaluations to further optimise the performance, as outlined in section 5.5. Due to time constraints each architecture and hyperparameter combination was only trained for 300 epochs

before stopping. However, in most cases convergence of performance was reached significantly before that point. The point of overfitting was hereby taken as the point of convergence, or optimum training. Overfitting was deemed to occur when an increase in the running average of the validation loss started to set in, whilst the training loss was continuing to decrease. The network weights from this point were then chosen for testing. In very few cases, this point was reached at the end of the 300 epochs. In retrospect, in the interest of reducing the overall training time a smaller number of epochs could have been chosen with a likely similar outcome in terms of results.

5.4.1 Results

The following metrics were computed as comparative results: cavity Dice score, myocardium Dice score, combined cavity and myocardium Dice score and myocardium breakage fraction. The values for each network and metric can be found in table 5.1. Furthermore as a more task specific test set, ED and ES frames were extracted from the test set and tested separately, as the performance on such frames is of higher importance for the overall 2D to 3D pipeline.

5.4.2 Discussion

Based on the results in table 5.1 it can be seen that when using Bai et al.'s method, Dice scores similar to their own results in [204] could be achieved. The myocardium breakage fraction, however, was significant. Xu et al.'s network outperformed that of Bai et al. in most categories, except for myocardium Dice scores. Both networks were themselves, however, outperformed by the UNET implementation which was dominant for all Dice metrics and the myocardium breakage. It was therefore decided to take this network forward and optimise the performance of the UNET.

Examples of some segmentations using the UNET architecture can be seen in Figure 5.4. Examples of segmentations with myocardium breakages can be seen in Figure 5.7.

Table 5.1: Comparison of architectures for 2ch LAX segmentation. Networks based on architectures by Bai et al. [204], Xu et al. [269] and a proprietary UNET were tested and compared. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for all, ED&ES, ED and ES test cases.

Bai et al.	**All**	**ED & ES**	**ED**	**ES**
Combined DICE	0.948	0.942	0.950	0.931
Cavity DICE	0.941	0.936	0.951	0.907
Myocardium DICE	0.885	0.862	0.869	0.852
Myocardium Breakage	0.385	0.350	0.350	0.350
Xu et al.	**All**	**ED & ES**	**ED**	**ES**
Combined DICE	0.967	0.961	0.969	0.950
Cavity DICE	0.961	0.957	0.970	0.932
Myocardium DICE	0.883	0.862	0.870	0.852
Myocardium Breakage	0.212	0.193	0.213	0.173
UNET	**All**	**ED & ES**	**ED**	**ES**
Combined DICE	0.971	0.967	0.973	0.959
Cavity DICE	0.964	0.960	0.974	0.933
Myocardium DICE	0.894	0.876	0.885	0.865
Myocardium Breakage	0.112	0.113	0.130	0.096

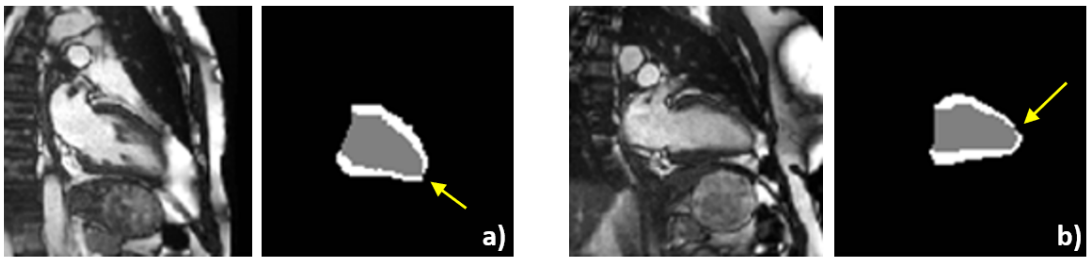


Figure 5.7: Examples of segmentations which contain a breakage in the myocardium segmentation, as pointed to by the yellow arrows. For most cases this occurs around the apex where the myocardium is thinnest, as seen in a), however, breakages can also occur further up the z-axis.

5.5 Segmentation UNET Optimisation

5.5.1 Introduction

To further improve the performance of the UNET and test its robustness as the main 2ch LAX segmentation network three approaches were used. Firstly, a hyperparameter optimisation was performed with the aim of fine-tuning the network to the task. Secondly, the UNET was tested for robustness by using Dr Lukaschuk's

Table 5.2: UNET Hyperparameter list and associated value ranges. Baseline values are given in bold.

Parameters	Value Ranges
Learning Rate	[10^{-2}, 5×10^{-3}, $\mathbf{10^{-3}}$, 5×10^{-4}, 10^{-4}]
Number of Layers	[2, 3, **4**, 5, 6]
Number of Convolutions per Layer	[1, **2**, 3]
Filter Scaling Factor	[1, 2, 3, **4**, 5]
Output Labels	[**Cavity & Myocardium**, Cavity & Epicardium]

data as an additional test set. Thirdly, the UNET was re-trained with the addition of Dr Lukaschuk's data and tested on the original test set to check if segmentation performance could be boosted by the addition of extra training data. Hereby, Dr Lukaschuk's data underwent the same data augmentation as the other data.

5.5.2 Hyperparameter Optimisation

Hyperparameter optimisation was performed across a list of parameters and associated values, as given in table 5.2. For each hyperparameter, the network was trained and tested separately, with all other parameters assuming their baseline values, as indicated in bold in the value ranges. For the parameter "Filter Scaling Factor" the integers represent the values of a scaling factor that is applied to the number of filters in the first layer of network which is 4. The number of filters for subsequent layers is then doubled for each layer in the encoding branch of the UNET. The number of filters is reversed for the decoding branch. E.g. A filter scaling factor of 2 for a network with 3 layers gives the following filters: $[2 \times 4 = 8, 2 \times 4 \times 2 = 16, 2 \times 4 \times 2 \times 2 = 32]$, i.e. [8, 16, 32] filters in the encoding branch and the opposite in the decoding branch. Similar to Xu et al.'s network, it was also of interest to trial different types of output labels. Using cavity & epicardium meant that myocardium labels had to be calculated afterwards.

5.5.3 Results

The results of the segmentation UNET hyperparameter optimisation can be found in Figure 5.8. Experiments A-E investigated the learning rate, the number of layers,

Table 5.3: Testing the previously trained UNET on data from Dr Lukaschuk. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for ED&ES, ED and ES cases.

Robustness Test	**ED & ES**	**ED**	**ES**
Combined DICE	0.953	0.960	0.946
Cavity DICE	0.931	0.949	0.913
Myocardium DICE	0.746	0.727	0.765
Myocardium Breakage	0.192	0.230	0.155

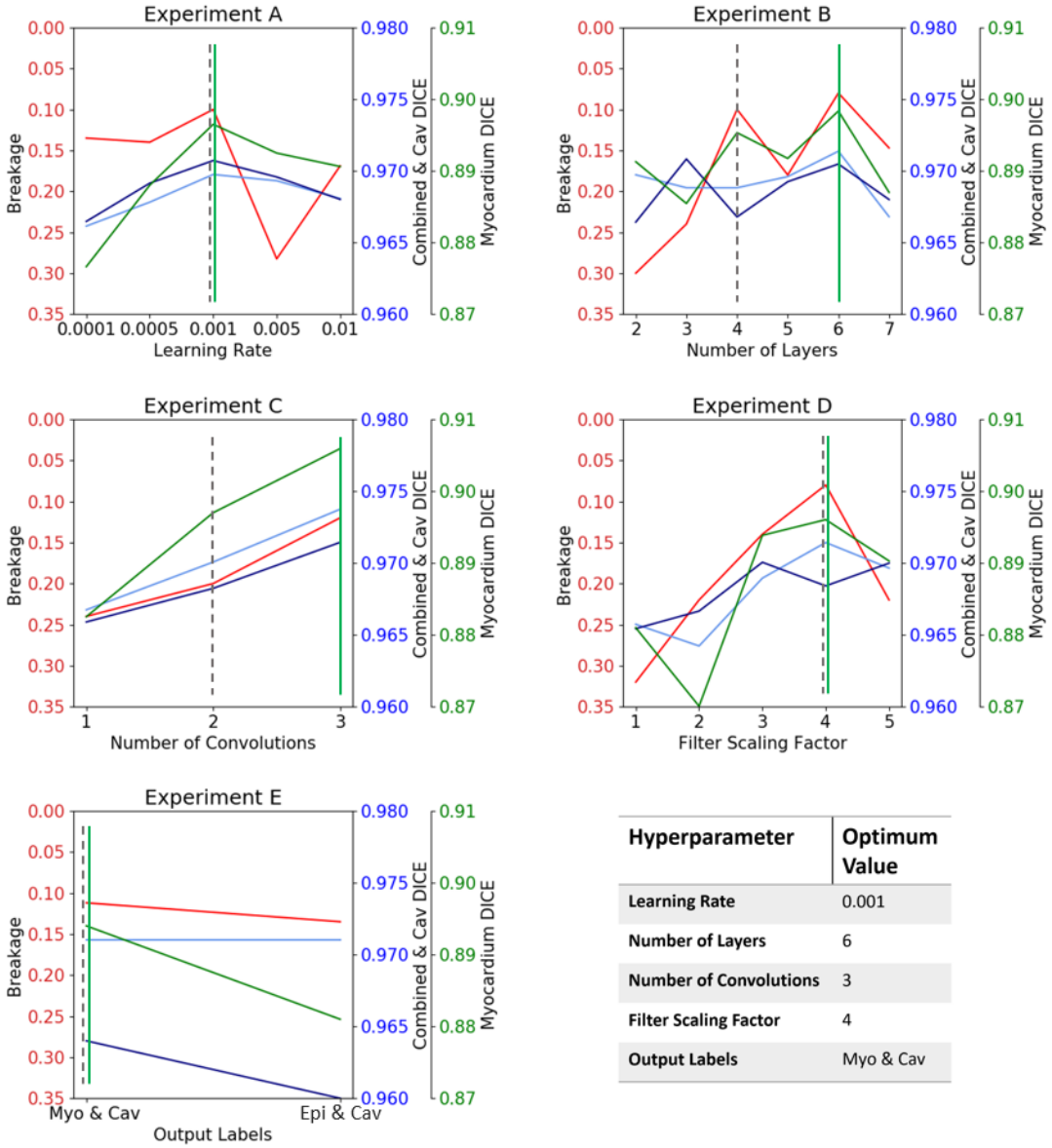
Table 5.4: Re-training UNET with additional data form Dr Lukaschuk and testing on original test set. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for all, ED&ES, ED and ES cases.

Joint Training Test	**All**	**ED & ES**	**ED**	**ES**
Combined DICE	0.972	0.968	0.972	0.964
Cavity DICE	0.961	0.953	0.967	0.929
Myocardium DICE	0.890	0.869	0.878	0.852
Myocardium Breakage	0.092	0.110	0.126	0.094

the number of convolutions per layer, the filter scaling factors and the type of output labels respectively. Furthermore, table 5.3 displays the results of the robustness test where the previously trained UNET was tested on Dr Lukaschuk's data. Dice scores for combined cavity and myocardium, just cavity and just myocardium, as well as the myocardium breakage fraction are displayed for ED&ES, ED and ES cases. Lastly, table 5.4 displays the results for the re-trained UNET when using both the original and Dr Lukaschuk's data for training and testing on the original data set.

5.5.4 Discussion

UNET hyperparameter optimisation has shown some clear trends in network performance for the investigated variables as seen in Figure 5.8. For the learning rate, as seen in previous sections, very high values can often lead to decreases in performance, as the network ends up bouncing around the local minimum without properly converging. Equally low values will prevent the network from reaching convergence. The baseline value of 10^{-3} seems to provide optimum validation results, both in terms of Dice scores, as well as for the myocardium breakage fraction. The



Hyperparameter	Optimum Value
Learning Rate	0.001
Number of Layers	6
Number of Convolutions	3
Filter Scaling Factor	4
Output Labels	Myo & Cav

Figure 5.8: Results of UNET hyperparameter optimisation experiments A-E. The red line indicates the fraction of test cases that showed a breakage in the myocardium segmentation (Note that the axis is flipped upside down to display low breakage values at the top of the graph). The light blue line indicates the Dice score of the combined cavity and myocardium labels, whilst the dark blue and green lines indicate the Dice scores of cavity and myocardium labels respectively. The gray dashed vertical line indicates the baseline parameter, whereas the green solid line indicates the best performing parameter value. The attached table gives a summary of all optimum values.

network employed an Adam optimizer which calculates learning rates individually for each network weight based on estimates of the first and second moments of the gradients. This means that the learning rate used in this network is as such adaptive. Nonetheless the initial learning rate does still matter, as it forms an upper limit for the calculation of the individual rates. Varying this limit during training by using a form of learning rate decay such as exponential decay could help to accelerate overall training whilst minimizing problems of convergence later in training which would otherwise be caused by high learning rates.

Network performance seems to improve with an increasing number of layers. Dice scores and myocardium breakage values are mostly in agreement on this, particularly globally, however, there are fluctuations and small differences in local trends between the metrics. The increase in performance with depth was expected, as more layers mean an increase in learnt parameters and network capacity which, as long as sufficient data is available, usually results in better performance. However, as seen in Figure 5.8, as the number of layers exceeds 6, network performance seems to drop, as universally indicated by the metrics. This might be related to the fact that there is a limit after which the preservation of local segmentation features becomes difficult, given the reduction in spatial dimensionality with each layer. Overall an increase from the baseline of 4 to 6 layers might bring better network performance.

The number of convolutions shows a clear upwards trend in performance, as indicated by all metrics, suggesting that large gains in performance can be achieved with increasing numbers of convolutions per layer. However, due to computational constraints, no larger numbers than 3 convolutions could be tested.

An increase in the filter scaling factor also seems to have a positive effect on network performance with an optimum scaling factor of 4. This is equivalent to 16 filters in the first layer, with filter numbers doubling each layer after that. Network performance seems to drop off with high scaling factors though. Training curves have indicated early levels of overfitting suggesting that more data would be needed to profit from including more filters.

The re-training with cavity and epicardium labels has not shown any improvement in the results. The combined Dice scores are much the same compared to the previous results in table 5.1, while the cavity and myocardium Dice scores are significantly worse. Contrary to initial expectations the network was able to learn myocardium and cavity labels more efficiently.

Overall, when comparing the influence of different parameters on network performance it can be argued that the number of convolutions and number of filters seem to be more dominant than the depth of the network. Assuming that enough images are available for training, which can be helped by sufficient augmentation, increases in these parameters will likely yield better performance. While this is not particularly surprising, it is interesting that the network depth is less of an important factor, particularly as segmentation is a problem that is strongly related to resolution. The impact of the learning rate resonates with common CNN behaviour in the literature.

Interestingly, the myocardial breakage fraction only once falls below the 0.1 mark, meaning that for most hyperparameter combinations, more than 10% of cases display a break in the myocardial segmentation. This is a non-negligible amount which poses a problem, particularly when considering further downstream processes in the 2D to 3D transformation pipeline. In order to facilitate easy registration between different planes, a single non-interrupted myocardium is necessary and breakages should be minimized.

Testing the trained UNET on Dr Lukaschuk's data, with the results given in table 5.3, has shown that the network is very sensitive to the applied segmentation method. As previously explained and shown in Figure 5.2, Dr Lukaschuk's data was generated using an interpolation method, leading to an over-prediction of the myocardium and cavity labels, effectively increasing the cavity size and pushing the myocardium outwards. As seen in table 5.3, when testing on data from this set, a significant drop in Dice scores compared to the original UNET Dice scores in table 5.1 can be observed. This is particularly true for the myocardium Dice scores which drop from an average of 0.894 to 0.746 for the whole test set. Myocardium Dice being

affected the most is reasonable, due to the relatively lower number of myocardium pixels, meaning that if there is a systemic shift in the segmentation, the percentage difference will be greatest here. The Dice scores shown here are significantly lower than what has been achieved in the literature and the segmentations are unlikely to be useful for downstream processes which require high levels of registration accuracy.

However, it was of interest to test whether using Dr Lukaschuk's data as additional training data would help optimise the network performance on the original test data set, i.e. not including test data from Dr Lukaschuk's set. Results in table 5.4 show that there is a marginal increase in Combined Dice score compared to the UNET test in table 5.1, yet both cavity and myocardium Dice experience a reduction in scores. It is reasonable to argue that the performance boost the network would have received due to the increase in training data has been offset by the systemic difference in ground truth segmentation sets. On a combined Dice level, these two factors seem to balance each other out, though for cavity and myocardium labels, the inclusion of the additional data seems to lead to more confusion rather than an improvement in scores. It must, however, be noted, that the myocardial breakage has improved compared to the initial UNET test, with an overall breakage fraction of 9.2% compared to 11.2%. Whilst this is still an unsatisfying percentage for test cases displaying myocardium breakage, the addition of more training data has shown to mitigate this issue slightly. Generally, when combining data from different source difficult choices arise. One the one hand it is desirable to have algorithms with a high level of generalisation, i.e. not having to depend on a specific segmentation network for a task. However, on the other hand it might be more valuable to work with a consistent data set if that data set has a certain interpretation of how the structure should be segmented that is different from others. This would make it easier for the algorithm to learn its task well. In the case in this chapter the results suggest that an inclusion of the extra data does not majorly affect segmentation results, however, in general the decision of whether to include external data depends on the quality and variability of the data set.

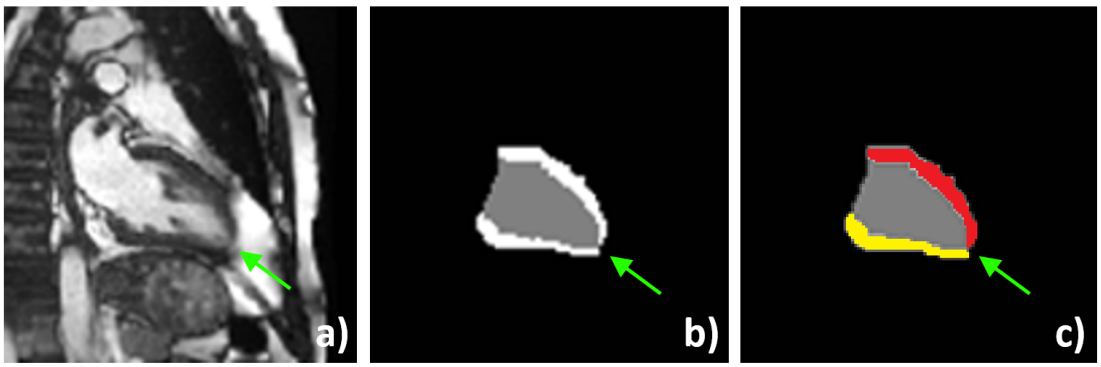


Figure 5.9: Example of a segmentation which contains a breakage in the myocardium segmentation, as pointed to by the green arrows. For most cases this occurs around the apex where the myocardium is thinnest. a) Original CMR image b) segmentation with breakage at apex c) breakage resulting in two distinct pixel areas and labels for myocardium (red and yellow).

5.6 Addressing Myocardial Breakage

5.6.1 Introduction

In the previous section it was shown that myocardial breakage is a non-negligible problem. As seen in the example in Figure 5.9, a breakage in the myocardium label often occurs around the apex for ED frames where the myocardium is at its thinnest and the segmentation network fails to assign a continuous area of pixels to the myocardium label, leading to two distinct pixel regions and labels for the myocardium. This break is anatomically incorrect and can cause issues further down the line in the 2D to 3D transformation pipeline. To address this issue, it was hypothesised that a cGAN structure might be more applicable in the segmentation process. Ideally, the additional adversarial loss in the cGAN would penalise myocardial breakages in the segmentations as unrealistic results, given that the training data does not feature them. It would therefore help push the network in a direction to minimise their occurrence, whilst maintaining a high reconstruction accuracy throughout.

5.6.2 Methods

To test this hypothesis, the UNET architecture was integrated as a generator into the pix2pix structure applied in chapter 4. The discriminator structure and network hyperparameters were directly transferred as well. The joint training set, as outlined in the previous section, was used for training. Data splits were used as before, with testing and validation data stemming from the original data set. Similarly the same metrics as before were applied, namely combined, cavity and myocardial Dice scores, as well as myocardial breakage fraction.

Training the cGAN largely followed the strategy applied in Chapter 3. Due to the lack of a straightforward convergence point, using overfitting to determine optimum training is ill-suited. Instead all cGANs were trained for a given number of epochs and validation performance was tested at constant intervals throughout the training process, i.e. after every epoch. The network weights for the top 3 validation performances were then saved and later used for testing. The results from these were then averaged to provide the overall test result for the underlying experiment. This helps reduce the effect of one-off outlier values and forms a compromise between giving the network enough training time to achieve high quality results and making the experiments feasible to execute. Similarly to before, 300 epochs was chosen as the optimum value for this.

The cGAN was initially trained with all the baseline parameters. It was then decided to undertake a hyperparameter optimisation. Two different parameters were investigated in their effect on network performance, namely the weighting of the adversarial loss and the use of patches instead of full images in the discriminator.

For the baseline experiment, the adversarial loss was given an initial weighting of 0.01 in the overall loss, with the reconstruction loss taking a weighting of 1, and full images were used instead of patches. For the hyperparameter optimisation experiments, the range of weighting values that was tested was [0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 0.1], where the values represent the adversarial loss weighting, with the reconstruction loss weighting remaining at 1

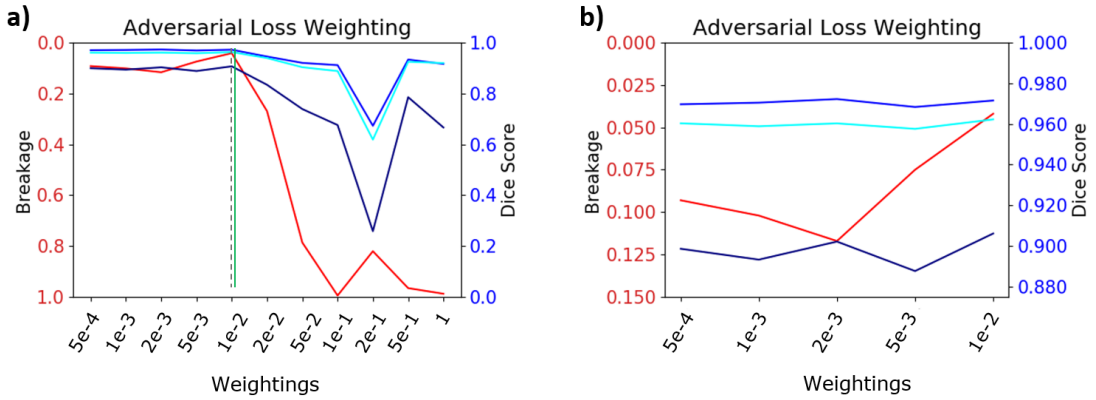


Figure 5.10: Results of cGAN for 2ch LAX segmentation using different adversarial loss weightings. a) Displays the Dice scores for combined myocardium and cavity (blue), cavity (light blue) and myocardium (navy blue), as well as the myocardium breakage fraction (red) for 11 different weighting factors. b) is a zoomed version of a) focusing on just the first 5 weighting factors. Note that the y-axis for Breakage is flipped upside down to display low breakage values at the top of the graph. The gray dashed vertical line indicates the baseline parameter, whereas the green solid line indicates the best performing parameter value.

throughout. For the patch discriminator, i.e. the PatchGAN, patches of 64x64 pixels in dimension were trialled and compared to the non-patch baseline performance.

5.6.3 Results

The results of the baseline cGAN performance and the PatchGAN can be found in table 5.5. The results of the adversarial loss weightings hyperparameter optimisation can be found in Figure 5.10. The right hand side graph is hereby a zoomed version of the left graph.

5.6.4 Discussion

The results in table 5.5, when compared to previous results in table 5.4, show that using the cGAN setup has not had a significant impact on the combined Dice score, but has improved the cavity Dice score and particularly the myocardium Dice scores by a non-negligible amount. Myocardium Dice scores have improved from 0.890 to 0.906, whilst cavity Dice scores changed from 0.961 to 0.962 for the whole test set. Furthermore, and most interestingly given the hypothesis of this section, the

Table 5.5: Segmentation results of the cGAN when using baseline parameters and when using a PatchGAN setup. Dice scores for combined myocardium and cavity, just cavity and just myocardium, as well as the fraction of cases that show a breakage in the myocardium segmentation are shown for all, ED&ES, ED and ES cases.

cGAN Baseline	**All**	**ED & ES**	**ED**	**ES**
Combined DICE	0.971	0.967	0.973	0.961
Cavity DICE	0.962	0.957	0.970	0.932
Myocardium DICE	0.906	0.884	0.895	0.862
Myocardium Breakage	0.042	0.054	0.060	0.048
PatchGAN	**All**	**ED & ES**	**ED**	**ES**
Combined DICE	0.972	0.969	0.973	0.962
Cavity DICE	0.959	0.953	0.968	0.934
Myocardium DICE	0.898	0.880	0.890	0.859
Myocardium Breakage	0.048	0.057	0.063	0.051

myocardium breakage fraction has reduced form 9.2% to 4.2% for the whole test set. This represents a reduction in broken myocardium segmentations of $\sim 54\%$.

Analysing the results in Figure 5.10 shows that cGAN performance is highly sensitive to the adversarial loss weightings. Low weightings in the region of <0.005 produced myocardium breakage fractions that are similar to those that have been seen for the optimised UNET. Dice scores remained relatively stable for the weighting factors in this range with no obvious trends being apparent. The myocardium breakage fraction reached an optimum value at an adversarial loss weighting of 0.01 which coincidentally was chosen as the baseline value. There is a slight rise in myocardium Dice scores for this value which is given in table 5.5. The addition of the adversarial loss not only ensures a lower breakage of the myocardium, but by doing so also has a positive effect on the overall segmentation accuracy of the myocardium. Beyond this point, the performance does, however, seem to have broken down. The increased weighting of the adversarial loss, caused less weighting to be placed on the reconstruction loss, thereby resulting in poorer overall segmentations. This in turn affected the rate of breakage of myocardiums in the segmentations. When analysing cGAN loss functions at such higher weighting levels, very unstable training performances could be observed. It has to be noted though that cavity Dice scores, and by extension combined Dice scores, mostly remained at

a level where a large majority of the cavity area was correctly labelled, frequently producing scores in the low 90s to upper 80s. This, however, is insufficient for reliable downstream use of the system.

Table 5.5 also displays the results for the PatchGAN test, where instead of full images, image patches of 64x64 pixels where fed to the discriminator. Theoretically this allows the network to focus on local features more closely with the aim of reducing myocardium breakage even further. However, the results in table 5.5 seem to indicate that this network adaption did not have a substantial effect on its performance. If anything small reductions in Dice and breakage scores can be observed.

5.7 Conclusion

As a first step of creating a 2D to 3D volume reconstruction pipeline, 2ch long-axis segmentation is vital. The work in this section has shown that such a segmentation network can be built for UK Biobank data and reach a level of accuracy that surpasses the previous state-of-the-art on this task.

Whilst the segmentation method performed well on the initial test set in terms of Dice scores, several issues were identified. Firstly the number of test cases that displayed a break in the myocardium segmentation where none should exist, was non-negligible. Secondly issues relating to the smoothness of the myocardium contours in the computed segmentations existed. These were, however, to some degree related to the lack of smoothness in the ground truth data which itself was related to the pixel-based method of segmentation. Thirdly, the robustness of the network was low, as highlighted by its comparatively poor performance on a test set created by another operator. This demonstrated that the method of segmentation for creating test sets has a large influence on the performance of the network if it has been trained with data created with a different method.

To increase performance, several methods have been investigated. Hyperparameter optimisation of the UNET architecture showed potential for performance increases given the right hyperparameter choice. Using the additional data set as extra training data has also shown to boost network performance, reinforcing the notion that more training data of varying degrees of quality is usually better than small amounts of high quality data. Furthermore, by applying a cGAN architecture with an adversarial loss component, myocardium breakage can be reduced substantially by 54% whilst maintaining high Dice scores.

In the future, this method could be improved further by incorporating more ground truth data and augmentations for training. When using the cGAN approach, longer training times might also result in network weights with better performance. Additional improvements in network robustness and segmentation accuracy can likely be achieved by implementing a multi-view segmentation network. Hereby multiple inputs and outputs corresponding to SAX, 2ch LAX and 4ch LAX views

would be trained together, allowing the network to share features between the views rather than treating them independently. Ensembles of segmentation networks could also be implemented to improve performance. Hereby multiple networks are trained and tested independently and a final segmentation output is generated by considering the outputs from all networks, according to an average or majority rule.

Lastly, in the field of medical image segmentation prior knowledge can often be the key to ensure anatomical correctness and prevent medically implausible artefacts like fragmented structures of pixel islands from occurring. Such prior shape knowledge can be introduced in the form of active/statistical shape models that learn representations of valid shapes or active contour models that employ deformable splines. Both of these methods are being more heavily used in combination with DL segmentation architectures, both implicitly and explicitly. Implicit methods allow end-to-end training, like Li et al. [270] who use a spatially encoded loss with a special shape attention mechanism, whereas explicit methods use the shape constraint as a post or pre-processing step, such as Medley et al. [271] who use an active shape model initialized with CNN generated features maps. In medical imaging where anatomical correctness is often key for downstream applications, shape-constraining methods are of high relevance and would be worth investigating in the future in relation to the application in this chapter.

The key contributions of this chapter are:

- Comparative evaluation of different deep learning architectures for automated 2ch LAX segmentation of UK Biobank CMR data.
- Development of optimised UNET for automated 2ch LAX segmentation of UK Biobank CMR data.
- Development of optimised cGAN for automated 2ch LAX segmentation of UK Biobank CMR data maximising Dice score based segmentation accuracy whilst minimising the fraction of cases with myocardial segmentation breakages.

6

3D Cardiac Function Modelling for MACE Risk Prediction

Contents

6.1 Introduction 196
6.1.1 Motivation 196
6.1.2 Overview of Methodology 197
6.2 Data 200
6.2.1 Introduction 200
6.2.2 Pre-processing 201
6.3 Part I: 3D Functional Modelling 203
6.3.1 3D Network Comparison 203
6.3.2 Hyperparameter Optimisation 208
6.3.3 Part I Discussion 211
6.4 Part II: MACE Classification 215
6.4.1 Classification Methodology Overview 215
6.4.2 Direct Linear Classification 216
6.4.3 Classification CNN with ED & ES Ground Truth Data . 218
6.4.4 Classification CNN with ES GT & Prediction Data . . . 221
6.4.5 Part II Discussion 226
6.5 Conclusion 229

6.1 Introduction

6.1.1 Motivation

Major Adverse Cardiovascular Events (MACE) are clinical endpoints that can occur in patients following severe cases of Myocardial Infarction (MI). Depending on the definition, these events can include heart failure, recurrent angina pain, cardiovascular death, and strokes which, if not treated properly, can result in fatal outcomes for patients [272]. To allow for the effective treatment of MI patients and thereby minimize the likelihood of such fatal MACE outcomes, accurate prediction and stratification of MACE risk is crucial. MACE risk prediction has long been a point of active research and several biomarkers, such as left ventricle ejection fraction (LVEF), are commonly used in clinical practise [2, 245]. However, while these form established parameters for assessing global LV function, they often fail to account for regional abnormalities in myocardial shape and function and it is widely accepted that such one-dimensional approaches lack patient-specific complexity [3, 273, 274].

Therefore, to facilitate a more holistic approach for MACE risk prediction, a thorough understanding of structure-function interactions in the heart is thought to be key. As seen in Chapter 4, learning structure-function interactions can be achieved by modelling cardiac function with generative networks, such as cGANs. These are ideally suited for learning underlying data distributions and extracting relevant features from such, an ability that makes them highly applicable for gaining a better understanding of patient-specific cardiac function.

When applied to 3D Cardiovascular Magnetic Resonance (CMR) data, a data source rich in structural and functional information, using such methods is expected to provide more patient-specific understanding of cardiac function. This in turn is hypothesised to enable more comprehensive MACE risk evaluation, as well as provide a more holistic understanding of pathophysiological, CMR-based imaging biomarkers associated with MACE risk in MI patients.

The aim of this chapter is therefore twofold: Firstly, to test the suitability of different generative approaches for modelling 3D cardiac function of MI patients.

This is done in order to learn MACE associated structure-function interactions, akin to the methods in Chapter 4. And secondly, to investigate the use of such networks and other methods in improving MACE risk prediction and providing further clinical understanding of MACE-related pathophysiology. Hereby it is worth noting that given the generic nature of the term "MACE" which encompasses a wide range of clinical outcomes, any derived pathophysiological understanding will have its limitations. This if further explored in the discussion.

6.1.2 Overview of Methodology

The work in this chapter is split into two parts, as depicted in Figure 6.1, which are aligned with the respective chapter aims.

Part I

The aim of Part I was to use a generative network to learn 3D structure-function interactions in hearts of MI patients by modelling cardiac function based on structural CMR input data. More specifically this means using 3D volume-to-volume translation networks for predicting ES LV segmentation volumes based on corresponding ED LV segmentation volume inputs. This allows for the modelling of cardiac function in a simplistic way and will subsequently facilitate the derivation of pathophysiological information of MI patients. For this purpose, several different network architectures were investigated, namely a 3D UNET, a 3D cGAN and a 3D VAE.

Each network was trained on pairs of ED and ES segmentation volumes from just non-MACE patients. Training on just non-MACE patients allowed the network to learn structure-function interactions specific to this particular patient group, so that during test time when both non-MACE and MACE cases are used, deviations in the predictions for MACE subjects can be attributed to MACE-related pathophysiological changes.

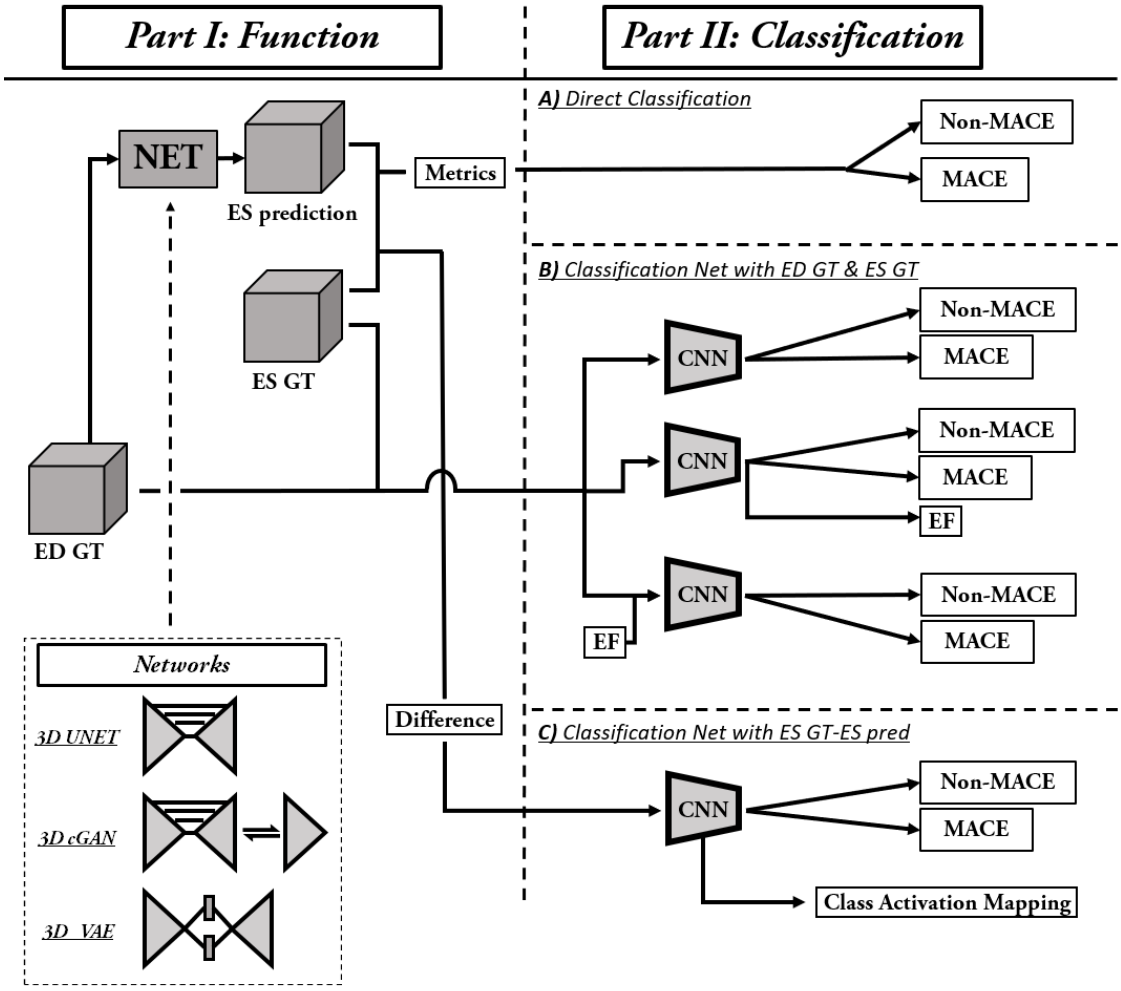


Figure 6.1: Overview of Methodology. Part I compares suitability of 3D UNET, 3D cGAN and 3D VAE architectures in predicting 3D ES volumes from ED volume inputs based on metrics including Dice Scores, Hausdorff Distance and EFs. Part II deals with MACE classification through a) direct classification given Part I metrics, b) a classification CNN given ES & ED ground truth data with additional EF input or loss and c) a classification CNN given the differences between ES predictions and ES ground truths. This is additionally used for Class Activation Mapping (CAM) to investigated MACE-related pathophysiology.

Part II

Part II focused on the prediction of MACE risk. The classification of patients into MACE and non-MACE outcome groups was hereby performed using three different strategies. Firstly, metrics derived from Part I including Dice Scores, Hausdorff Distances and Ejection Fractions were evaluated directly for MACE risk prediction. Secondly, a MACE classification CNN was trained and tested using a combination of ED and ES ground truth data. Alterations in the CNN's architecture including additional EF inputs and EF regression losses were also investigated. Lastly the classification CNN was re-trained using the difference between ES predictions and ES ground truth data for MACE and non-MACE cases from Part I and Class Activation Maps (CAMs) were implemented to facilitate a better understanding of MACE-related pathophysiology.

6.2 Data

6.2.1 Introduction

The data set upon which this chapter is based stems from two multi-center randomized trials, namely the AIDA-STEMI (Abciximab Intracoronary versus intravenously Drug Application in ST-Elevation Myocardial Infarction) [275] and TATORT-NSTEMI (Thrombus Aspiration in Thrombus Containing Culprit Lesions in Non-ST-Elevation Myocardial Infarction) [276] studies.

As part of these, 1235 patients suffering from either acute ST-Elevation MI (STEMI) or acute Non-ST-Elevation MI (NSTEMI) were imaged using an ECG-gated balanced steady-state free precession (b-SSFP) cine protocol to assess LV function and structural damage. During such, SAX and LAX scans were acquired for all patients. All STEMI patients were part of the AIDA study, whilst all NSTEMI patients were part of the TATORT study. More information regarding image acquisition protocols and the trials in general can be found in [275, 276].

Out of the original cohort 1021 cases (STEMI=723, NSTEMI=298) possessed corresponding 12-month follow-up data regarding their MACE outcomes. This data indicates a total of 73 MACE cases (STEMI MACE=52, NSTEMI MACE=21), comprising 20 cases of congestive heart failure, 21 cases of re-infarction and 32 cases of death [275, 276]. Four cases from the non-MACE STEMI group were excluded from further analysis, due to significant data disruption.

Scans from these patients were subsequently segmented using an automated UNET enabled segmentation method from [277, 278]. Based on the segmentations, two 3D triangulated meshes corresponding to the ED and ES myocardial shapes were constructed for each patient using smooth cubic Hermite interpolation as described in [279] and [280]. The resulting meshes were aligned by orienting them according to their SAX canonical position, as well as centering them on the meshes' centres of mass. A graphical summary of the data distribution can be seen in Figure 6.2.

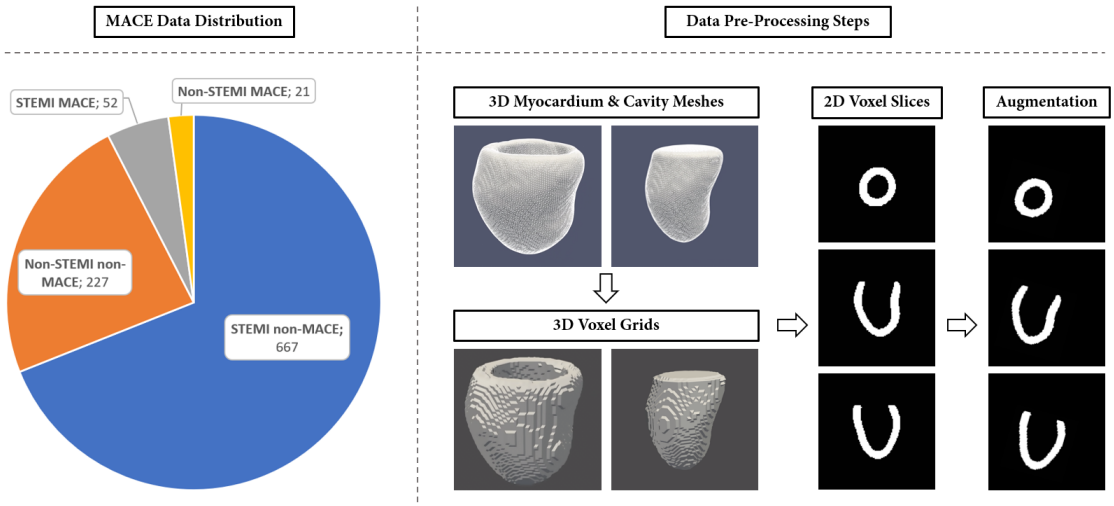


Figure 6.2: MACE data distribution and data pre-processing steps. These include transforming 3D myocardial and cavity meshes into 3D segmentation voxel grids. Volume augmentations in the form of random rotations and translations were then applied to increase the data set.

6.2.2 Pre-processing

The 3D mesh volume pairs were converted to multi-label segmentation volume grids whereby the voxel labels correspond to LV myocardium, LV cavity and background. During the conversion process a long-axis and short-axis voxel spacing of 1.4mm was chosen so to fit the meshes into a $128 \times 128 \times 128$ voxel 3D volume array. This size was chosen as a compromise between memory constraints and shape resolution, in accordance with relevant literature. To increase the size of the data set and improve data variety and network generalisation, random rigid augmentations were performed on-line on volume pairs during training. These included rotations of up to ± 90 degrees and translations of up to ± 10 voxels around and along all cartesian axes respectively. The number of augmentations was chosen to always ensure an equal distribution of non-MACE and MACE cases within the training set when both classes were trained on to prevent class imbalance.

For the generative network volume-to-volume translation in Part I, only non-MACE cases were used for training. Both non-MACE and MACE cases, however, were used for validation and testing. Due to the relatively low number of cases, MACE in particular, as well as the high variability within the data set, all training,

validation and testing was performed in a k-fold cross validation process, with k=5, from which results were averaged. This effectively allowed for all data to be tested on.

For each fold, the non-MACE data was split into training, validation and testing sets in a ratio of $\sim 7/1/2$ whereby the data from individual patients, including all its augmentations, belonged to only one of the groups, i.e. no train/test overlap. Each fold hence consisted of 680 non-MACE cases for training, 76 non-MACE cases for validation and 188 non-MACE cases for testing. The MACE cases were split into validation and testing folds in a ratio of $\sim 1/2$ with 25 and 48 MACE cases for validation and testing respectively. All training and validation cases underwent on-line augmentation to increase the data set to 6800 non-MACE cases for training and 380 non-MACE and MACE cases each for validation. For testing data no augmentation was used, leaving 188 non-MACE and 48 MACE cases for each fold. Due to the small number of MACE cases, this meant that some MACE cases were tested on multiple times across the 5 folds, however, as each fold network was trained with a different training data subset, this was deemed to be acceptable. A total 940 non-MACE and 240 MACE test outputs were thereby generated.

The test data outputs from Part I were then used as the new data set for network training, validation and testing in Part II, as is explained in more detail in the relevant section.

6.3 Part I: 3D Functional Modelling

In this section the aforementioned architectures for 3D functional modelling are outlined and their performance compared. Subsequently a hyperparameter optimisation of the best performing architecture was undertaken in an attempt to improve network performance further. Lastly, the average results for both non-MACE and MACE test groups were analysed qualitatively to gain a better understanding of MACE pathophysiology.

6.3.1 3D Network Comparison

Introduction of 3D Networks

The main generative network used for functional modelling was a 3D extension of the UNET architecture proposed by [66] which was configured with the aim of transforming 3D ED multi-label segmentation volumes into their corresponding ES counterparts, thereby modelling LV shape contraction. As seen in Figure 6.3, the architecture consisted of an encoding and a decoding branch with a bottleneck layer and skip connections spanning between them. In the encoding branch several layers of stacked blocks of 3D convolutions followed by batch normalisation, activation and max-pooling layers were connected back to back. The output was connected through the bottleneck to the decoding branch where several layers of stacked up-sampling and further convolutional layers were used to produce the output.

Each layer in the decoding branch was connected to outputs at respective depth in the encoding branch using skip connections. The final network layer consisted of a sigmoid activation which produced probability output volumes for each of the class labels, i.e. LV myocardium, LV cavity and background. For each output voxel, the label with the highest probability was chosen. During training, a reconstruction loss was computed by comparing the predicted ES output volumes to their gold standard using Dice loss as the metric.

As alternatives to the 3D UNET, a 3D cGAN and 3D VAE architecture were also investigated in their suitability for modelling 3D function. A 3D cGAN variant which incorporates the 3D UNET as its generator network was used. In addition to

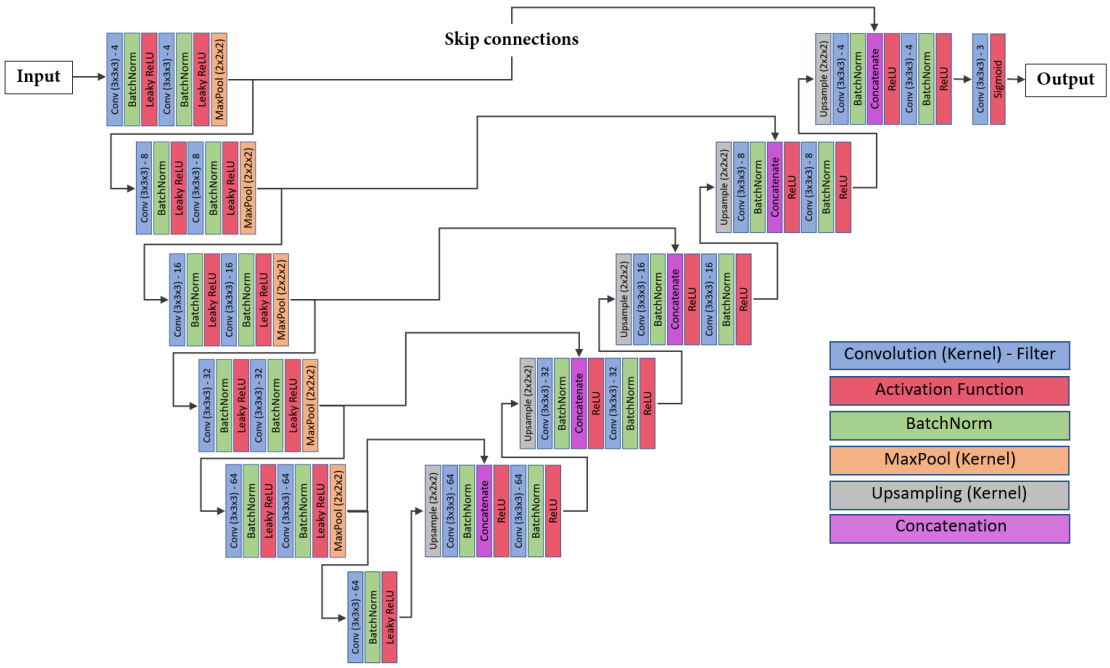


Figure 6.3: 3D UNET structure.

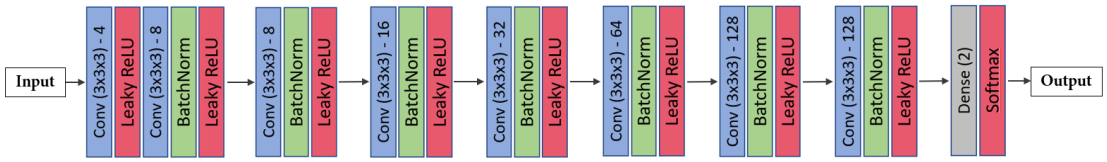


Figure 6.4: 3D cGAN Discriminator structure.

the Dice reconstruction loss of the generator the 3D cGAN also implemented an additional adversarial loss computed by a discriminator network. The discriminator followed the structure in Figure 6.4. No explicit hyperparameter optimisation was undertaken for the discriminator, but an established setup adapted from previous chapters was chosen instead.

A 3D VAE variant was also used which followed the general structure of the 3D UNET, however, at its bottleneck it included fully connected layers for the mean and variance vectors of the latent distribution. After using the reparameterization trick proposed in [92], the decoding branch of the 3DUNET followed, except for the lack of cross connections. Latent space dimensions for the 3D VAE were based on previous experience from chapter 3.

Table 6.1: Stage I performance results from 3DUNET, 3DcGAN and 3DVAE in prediction ES segmentation volumes based on ED segmentation inputs. Results are given for non-MACE (nM), MACE (M) and combined non-MACE and MACE (Comb) test cases for Combined Myocardium and Cavity Dice, Cavity Dice, Myocardium Dice, Hausdorff Distance and Absolute Difference in EF scores. The values in bold indicate the best value for each test group and metric among the three networks.

Metrics	Test Groups	3D UNET	3D cGAN	3D VAE
Combined Dice	nM	**0.913 ±0.032**	0.886 ±0.035	0.897 ±0.028
	M	**0.899 ±0.030**	0.862 ±0.039	0.883 ±0.026
	Comb	**0.910 ±0.032**	0.881 ±0.036	0.894 ±0.028
Cavity Dice	nM	**0.853 ±0.053**	0.833 ±0.050	0.837 ±0.050
	M	**0.830 ±0.055**	0.813 ±0.051	0.816 ±0.056
	Comb	**0.848 ±0.053**	0.829 ±0.050	0.833 ±0.052
Myocardium Dice	nM	**0.792 ±0.077**	0.756 ±0.065	0.773 ±0.065
	M	**0.756 ±0.078**	0.745 ±0.059	0.739 ±0.071
	Comb	**0.785 ±0.077**	0.754 ± 0.063	0.766 ±0.067
Hausdorff Distance	nM	**7.19 ±2.95**	7.51 ±2.53	7.41 ±2.49
	M	7.48 ±2.56	**7.46 ±2.29**	7.60 ±2.56
	Comb	**7.25 ±2.90**	7.50 ±2.45	7.45 ± 2.51
EF Absolute Difference	nM	**0.076 ±0.057**	0.082 ±0.059	0.081 ±0.060
	M	**0.115 ±0.092**	0.122 ±0.088	0.120 ±0.094
	Comb	**0.084 ±0.063**	0.090 ±0.065	0.089 ±0.068

3D Network Comparison Experiment

All three networks were individually trained on the non-MACE training data set using their baseline parameter values and tested on both non-MACE and MACE test data sets. During training, the validation set was used to evaluate the training progress. Early stopping triggers using persistent increases in validation losses over 5 epochs as indicators were used to identify optimum training points. Networks were trained for a total of 100 epochs, a compromise between sufficient time to achieve optimal training and time resources. All networks reached optimum training well before the final epoch. For each network, the mean and standard deviation for each of the applied metrics were calculated and averaged across all testing folds. These were then used to judge the networks' ability to accurately predict ES segmentation volumes and choose the best suited architecture.

Table 6.2: P-values for difference in performance between non-MACE and MACE test sets in table 6.1. Calculated using the Kolmogorov-Smirnov test.

***p*-VALUES**			
Metrics	**3D UNET**	**3D cGAN**	**3D VAE**
Combined Dice	$\mathbf{p<10^{-2}}$	$\mathbf{p<10^{-2}}$	$\mathbf{p<10^{-2}}$
Cavity Dice	$\mathbf{p<10^{-2}}$	$\mathbf{p<10^{-2}}$	$\mathbf{p<10^{-2}}$
Myocardium Dice	$\mathbf{p<10^{-3}}$	$\mathbf{p<10^{-2}}$	$\mathbf{p<10^{-3}}$
Hausdorff Distance	p=0.33	p=0.51	p=0.37
EF Absolute Difference	$\mathbf{p<10^{-3}}$	$\mathbf{p<10^{-3}}$	$\mathbf{p<10^{-3}}$

3D Network Comparison Results

Performance results for the 3D UNET, 3D cGAN and 3D VAE in predicting ES segmentation volumes based on ED segmentation inputs are given in table 6.1. These include the test set Dice scores for myocardium, cavity and combined myocardium and cavity labels, the Hausdorff Distance and the absolute EF difference for both MACE and non-MACE test cases individually, as well as the combined set of non-MACE and MACE cases. The KS test was used to ascertain the statistical significance of the difference in performance between non-MACE and MACE test sets by calculating the associated p-values. Values $< 10^{-2}$ were taken as being statistically significant. These can be found in table 6.2.

3D Network Comparison Discussion

The comparison of networks in table 6.1 has shown that in terms of absolute combined scores the 3D UNET outperformed the 3D cGAN and 3D VAE in all categories. This means that for a combined test set of non-MACE and MACE test cases, the 3D UNET was able to most accurately predict ES myocardium and cavity segmentation labels based on a ED volumetric input. Apart form the average Hausdorff Distance, the 3D VAE performed next best, with the 3D cGAN coming in last. The difference in performance between the 3D VAE and 3D cGAN,

however, was on average smaller than between the 3D UNET and the 3D VAE. Whilst absolute scores are a good indicator of general performance accuracy of the network, the secondary aim of this experiment was to learn structure-function interactions that are specific to MACE cases and cannot be found in a non-MACE population, as a way of differentiating between the two sub-populations. Therefore, the difference in performance between the two test groups is also of importance. Apart from the Combined Dice score, the absolute and percentage difference between the non-MACE and MACE test group results is the largest for the 3D UNET among all three networks. As outlined in table 6.2, all networks achieve a p-value of less than 10^{-2} for all metrics except Hausdorff Distance, thereby indicating an overall strong statistical significance of the differences between non-MACE and MACE results. Reasons for the difference in HD not being statistically significant are likely related to the inherent weakness of HD as a metric which is its sensitivity to individual outliers. HD was mainly included as it is a commonly used metric, however, mean contour distance might have been a more informative choice.

Reasons for the relatively superior performance of the 3D UNET compared to the other networks might lie in the fact that compared to the 3D VAE the 3D UNET contains skip connections which help maintain local features more easily. This, however, is usually more important when dealing with image rather than segmentation data, as the whole range of gray levels is used and blurring is much more of an issue.

The cGAN has proven to be a highly effective network for modelling function in previous chapters. However, in this application the additional adversarial loss seems to be of less use, as the problem of creating a "realistic" looking output is less of an issue with segmentation data compared to when dealing with gray scale image data. Therefore, the adversarial loss is likely not providing as much of a benefit to the training process as it was before. In fact it even seems to have added more confusion to the network and thereby hampered its performance.

Table 6.3: 3D UNET hyperparameters and associated value ranges. Baseline values are given in bold.

Parameters	Value Ranges
Number of Layers	[2, 3, **4**, 5, 6]
Number of Convolutions per Layer	[**1**, 2, 3]
Filter Scaling Factor per Layer	[**1**, 2, 3]
Dropout Fraction	[**0**, 0.2, 0.35, 0.5]
Learning Rate	[10^{-2}, 5×10^{-3}, $\mathbf{10^{-3}}$, 5×10^{-4}, 10^{-4}]

6.3.2 Hyperparameter Optimisation

Based on the previous comparison of network performance it was decided to take the 3D UNET forward and perform a hyperparameter optimisation. The aim of this was to both test the robustness of the network to changes in hyperparameters and to improve absolute network prediction scores whilst maintaining a statistically significant difference between non-MACE and MACE test set results.

Methods of optimisation

Two different methods of optimization were undertaken. Firstly, a single parameter was tested at a time to identify the effect of a change in said parameter on network performance. Hereby a range of values that were deemed suitable, based on appropriate literature and memory constraints, were tested sequentially with all other parameters remaining the same. For each parameter setting, k-fold cross validation was performed, where $k = 5$, and average results were computed.

The choice of optimisable parameters was based on their likely impact on training performance, based on experience from previous chapters, and observed consistency in the literature. This means that some parameters like the β values for the Adam optimizer which have been researched extensively in the literature were deemed not necessary to optimise any further. However, parameters such as network depth which can usually have a significant effect on training performance and are highly dependent on the underlying application, warrant optimisation.

The hyperparameters to be tested and their corresponding value ranges are given in table 6.3.

Similarly to the use in previous chapters, the Filter Scaling Factor (FSF) is the scalar by which the number of filters for each convolution in the network is scaled by. The number of filters per convolution in the encoding branch can hence be calculated using $N = 2^{(LayerNumber-1)} \times 4 \times FSF$, starting with $LayerNumber$ 1 and 4 being the smallest possible filter size. Reverse filter numbers were used on the decoding branch.

The network training process and hyperparameter comparison were based on the average performance of the validation sets for each fold. Early stopping triggers were used to identify the point at which optimum validation performance has been reached and training data overfitting started to occur. If the 3-epoch moving average of the validation loss was higher than its minimum score for 5 consecutive epochs, training was stopped, and the point of minimum loss was taken as the optimum. Parameter combinations were trained for a maximum of 50 epochs, a compromise between having sufficient time to reach optimum training and time constraints. In most cases, signs of overfitting started to occur before the final epoch.

As a second strategy, random combinations of hyperparameter values within the given ranges were tested in order to identify a more global optimum without having to use extensive grid-based searches. To aid computational efficiency whilst also maintaining training data variance, k-fold cross validation with k=3 was used. A total of 25 different random parameter combinations were tested which formed the maximum number of combinations possible given time and computational constraints.

Optimisation Results

The results for the first strategy of the hyperparameter optimisation process can be found in Fig. 6.5 which depicts the average validation losses across all folds for each tested hyperparameter value with all other parameters taking their baseline values.

The second strategy tested 25 different random hyperparameter value combinations, taking values from the respective discrete value ranges. The lowest validation loss of 0.0137 was achieved by a network with 5 layers, 3 convolutions per layer, a filter scaling factor of 2, a dropout fraction of 0.2 and a learning

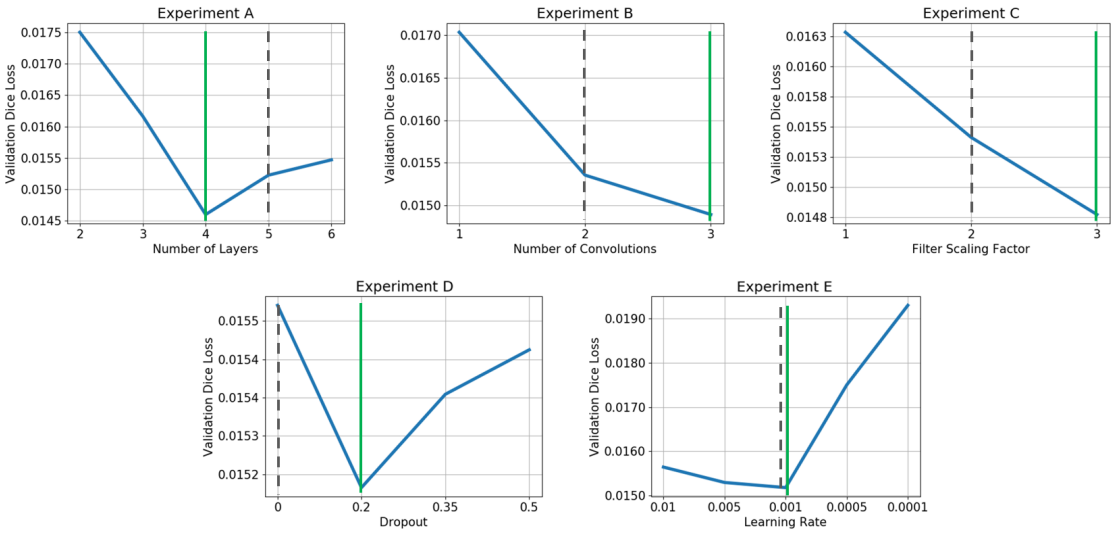


Figure 6.5: 3D UNET Hyperparameter Optimisation Results. Validation loss results for different numbers of layers, numbers of convolutions, filter scaling factors, dropout levels and learning rates are shown. The vertical grey, dashed lines indicate the baseline value for each parameter whilst the vertical green, solid lines show the optimum values.

rate of 10^{-3}. This network was then used as the "optimum" network and tested again. The test results for performance metrics and associated p-values for this networks are given in table 6.4.

Discussion of Optimisation Results

Hyperparameter optimisation results in Figure 6.5 confirm anticipated trends for the number of layers, convolutions and filters, as similar behaviour has been observed in previous chapters. Increases in convolutions and filters beyond what has been tested was not possible due to computational resource limitations. The difference in validation loss between the best and worst hyperparameter value was highest for the number of layers, followed by the number of convolutions and lastly the FSF. These differences are multiple times larger than those seen for different dropout levels. Therefore, it can be argued that dropout is a negligible parameter in this application. Lastly the optimisation has re-affirmed previous trends that network training, if resource limited, is highly sensitive to learning rates.

Following network re-training with optimised parameters, test results have shown an improvement in all absolute scores for Combined, Cavity and Myocardium

Table 6.4: Optimised 3D UNET metric results and associated p-values.

Metrics	Test Groups	3D UNET Results	p-values
Combined Dice	nM	0.918 ±0.028	$p<10^{-2}$
	M	0.903 ±0.035	
	Comb	0.915 ±0.030	
Cavity Dice	nM	0.864 ±0.049	$p<10^{-2}$
	M	0.835 ±0.058	
	Comb	0.858 ±0.051	
Myocardium Dice	nM	0.802 ±0.072	$p<10^{-3}$
	M	0.768 ±0.079	
	Comb	0.795 ±0.074	
Hausdorff Distance	nM	7.20 ±2.65	p=0.34
	M	7.44 ±2.70	
	Comb	7.24 ±2.66	
EF Absolute Difference	nM	0.074 ±0.054	$p<10^{-3}$
	M	0.112 ±0.085	
	Comb	0.082 ±0.063	

Dice scores and absolute EF differences, though no improvement in Hausdorff Distances. The relative differences between non-MACE and MACE groups are similarly present with p-values indicating statistical significance for all metrics except Hausdorff Distance.

6.3.3 Part I Discussion

Qualitative Analysis of 3D Function

Following hyperparameter optimisation, the test set results were graphically compiled in Figure 6.6 which highlights the difference in the prediction accuracy between non-MACE and MACE test sets when applying the 3D UNET. For each test set, the average ES myocardium prediction and average ES myocardium ground truth were computed and transformed into 3D meshes using the marching cubes algorithm. These meshes were then overlaid onto one another to highlight regional differences. The green mesh represents the average ES prediction, whilst the white mesh represents the average ES ground truth. Images a)-d) and e)-h) represent the four long-axis viewpoint directions along the x-y grid for non-MACE and MACE

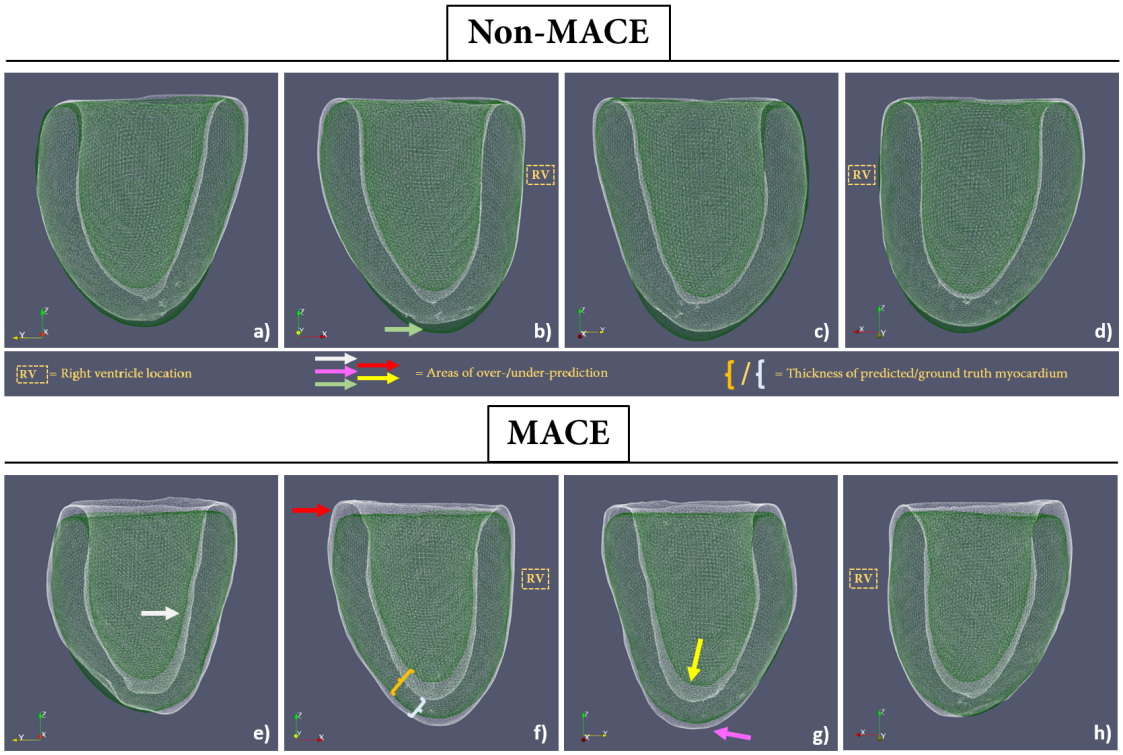


Figure 6.6: 3D mesh representations of 3D UNET prediction results. The averages of ES myocardium predictions (in green) and corresponding ES ground truths (in white) are shown for both non-MACE and MACE test sets along the four long-axis viewpoint directions. The arrows indicate regions of over-and underprediction for the non-MACE and MACE case. Using g) as an example, the endocardial cavity contour of the ES prediction is indicated by the yellow arrow. The endocardial contour of the ES ground truth sits below that, i.e. the line bounding the inner white mesh.

results respectively. Using image g) as an example, as it is most obvious here, the line pointed to by the yellow arrow represents the endocardial contour of the prediction, i.e. the cavity/myocardium border. Below that follows the endocardial contour of the ground truth with the epicardial contours of the prediction and ground truth following towards the apex of the volume respectively. In other words, the model is overestimating myocardial contraction in that region, meaning that both endo- and epicardial contours of the prediction are positioned closer towards the center of the LV.

The results highlight a few points already observed in the data in table 6.1. Primarily it shows an overall better agreement between predictions and ground truths in the non-MACE test set where divergence in contours is clearly less severe.

While non-MACE cases display a better global alignment, on closer examination it can be seen that an average longitudinal over-prediction near the apex does exist, as indicated by the green arrow in b). The ground truth exhibits more longitudinal contraction than the prediction. This is likely due to a few outlier cases in the test set, literally dragging the average down. In the MACE set, the situation at the apex is reversed. As indicated by the pink arrow in g), the prediction indicates a stronger contraction towards the LV center than the ground truth actually shows. This is somewhat expected, as myocardial longitudinal shortening has been associated with MACE [3, 281].

Next, the yellow arrow in g) indicates a significant over-prediction of the endocardial contraction in the MACE prediction compared to its ground truth. The predicted cavity volume is significantly smaller, correlating to much higher EFs than present in the ground truth. While a similar over-prediction also occurs for the non-MACE set, it is much less severe. This supports findings which indicate a strong correlation between lower LVEF & ES volumes (ESV) and MACE [273]. Similarly agreeing with the literature [273], the orange and white brackets seen in f) which indicate the myocardial thickness for ES predictions and ground truths respectively, show that myocardial thickness in MACE cases is reduced for ground truths compared to predictions.

The red arrow in f) also highlights an over-prediction of the longitudinal contraction at the base of the heart for MACE ES predictions compared to the ground truth. The base of the myocardium moves much further downwards in the predictions than it does in the ground truths which adds to a higher EF prediction. This does not seem to occur much in the non-MACE case. Lastly, the white arrow in e) indicates higher levels of mis-prediction in the anterior endocardial wall in the MACE cases compared to non-MACE cases. Similarly to the yellow arrow, this highlights a region where the predicted ES endocardium contracted more inwards than the ground truth.

Limitations & Improvements

The qualitative visualisation of results in Figure 6.6 is interesting to compare with the quantitative results in table 6.1 in so far that even though global differences, as assessed by Dice scores, are significantly different, it can be observed that divergences between ground truth and predictions are mostly confined to local regions. Analysing such regions independently would likely result in even higher levels of statistical significance. It highlights the fact that using Dice scores without any other form of assessment can be uninformative or even misleading.

During data acquisition, SAX slices at the apex have occasionally been missed, leading to a potentially incorrect estimation of myocardial geometry for some cases. However, this is thought to only be an issue for individual cases, rather than affecting the overall average. Therefore apical deviations can be taken as true differences between ground truths and predictions. In order to combat this issue in the future, however, LAX slices could be added into the mesh reconstruction to provide robustness.

General improvements to the method can include the use of larger data sets or at least higher levels of augmentation, trialling more hyperparameter combinations and increasing computational power to allow for larger filter sizes. Furthermore, hyperparameter optimisations could be run for the 3D VAE and 3D cGAN prior to comparing the networks. Given the improvement of the 3D UNET following hyperparameter optimisation it is difficult to say what the results of optimised VAE or cGANs would have been though.

Part I Conclusion

Overall it has been shown that similar to the methodology applied in chapter 4, 3D cardiac function can be modelled accurately using deep learning volume-to-volume translation networks. Moreover, and more interestingly, subject group specific function can be extracted by training on a sub-population of data, in this case that of non-MACE MI patients. Through this, differences in contractions for subjects who have experienced MACE could be highlighted.

6.4 Part II: MACE Classification

6.4.1 Classification Methodology Overview

Following the volume to volume translation in Part I, Part II addresses the MACE classification objective which was to investigate the use of results from Part I and other methods in their suitability for MACE risk prediction. The classification of test patients into MACE and non-MACE outcome groups was hereby performed in three separate ways, as outlined in Figure 6.1.

Firstly, classification was performed directly with a linear classification based on the metrics calculated from the output of the 3D UNET in Part I, namely the myocardium, cavity and combined myocardium and cavity Dice scores, Hausdorff Distance and absolute EF difference. The hypothesis behind this was that the network is more accurate at predicting non-MACE cases and thus the lower accuracy in MACE cases can be used to classify them.

Secondly, a separate classification network was trained using ED and ES ground truth data of both non-MACE and MACE subjects. The hypothesis was that a CNN would be well suited a extracting complex myocardial shape information in order to achieve a better classification accuracy than using isolated metrics. Hereby architecture alterations to further improve classification scores were also evaluated. These included using the ejection fraction as an additional network input parameter as well as using it as an additional output to create a secondary regression loss for network training.

Thirdly, the classification network was re-trained using differences between ES ground truth data and corresponding ES predictions from the 3D UNET for both non-MACE and MACE subjects. The hypothesis behind this was that by using the difference, the classification network would be able to focus on cardiac function instead of just ED and ES shape information, helping it improve its performance. Class activation maps were generated to highlight regions of attention in the network to help better understand MACE pathophysiology.

In order to quantify classification performance and compare strategies effectively, Receiver Operator Characteristic (ROC) curves were computed from which the Area Under the Curve (AUC) was calculated. Whilst this is straightforwardly done for the class prediction scores from the classification networks, the continuous metric values of the first strategy were normalised first to put them into a [0, 1] range. ROC curves were computed by picking equidistantly spaced thresholds along the respective data value range, classifying outputs at each threshold and computing the True Positive Rate (TPR) and False Positive Rate (FPR) to then be used in the ROC/AUC calculation.

6.4.2 Direct Linear Classification

Results

Results for the direct linear classification of patients into non-MACE and MACE groups based on metrics calculated in Part I can be seen in Figure 6.7. Hereby the metrics from the 940 non-MACE and 240 MACE test cases, as described in Part I, were used.

Discussion

The ROC curves in Figure 6.7 show the classification performance of using output metrics from Part I directly for linear classification into non-MACE and MACE categories. The EF ground truth (GT) is hereby taken as the baseline, with an AUC of 0.674. The figure clearly shows that none of the metrics derived from Part I can surpass this baseline performance. While table 6.2 showed that there is a significant difference in Dice scores between non-MACE and MACE test data sets, this difference was not sufficiently large to provide a basis for classification that outperforms using EF GT.

This highlights why LVEF is a commonly applied MACE biomarker in clinical practise. Nonetheless, it does show that the metrics do indeed have predictive power which validates the initial hypothesis. Though using these metrics for prediction would be sub-optimal and not recommended.

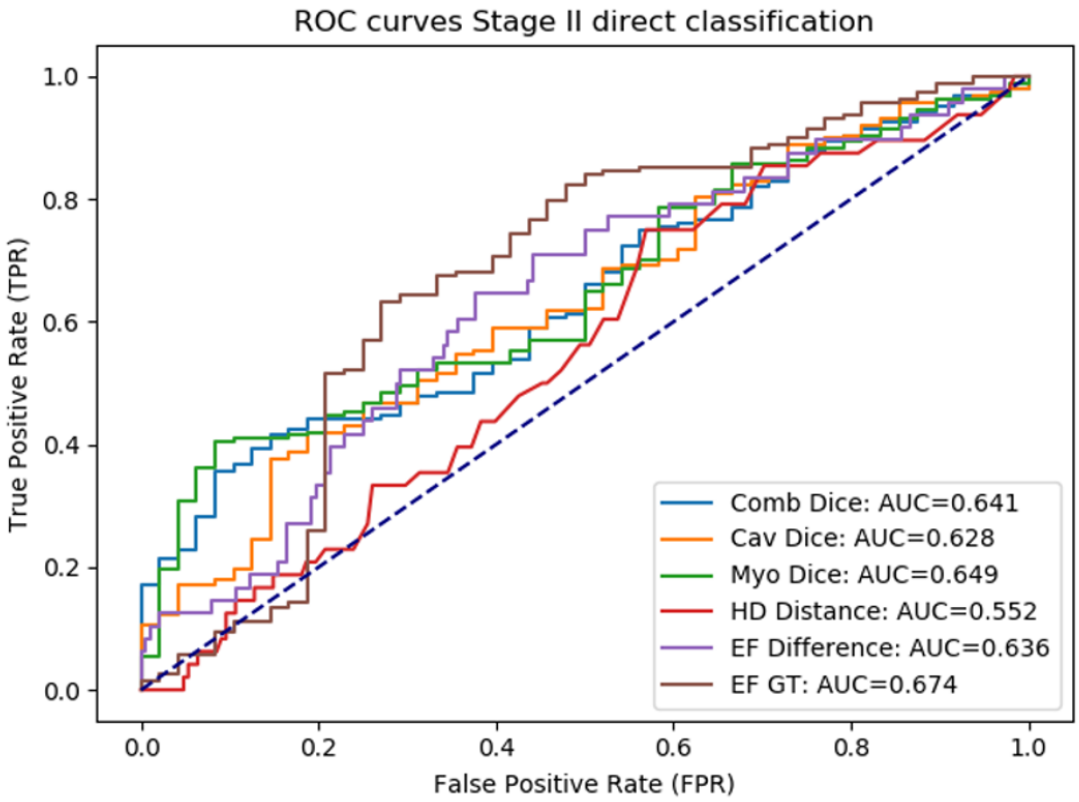


Figure 6.7: Receiver Operator Characteristic (ROC) curves for direct classification of non-MACE vs MACE subjects using different metrics derived from Stage I function modelling. The Area under the Curve (AUC) for each metric is given in the figure legend. Metrics include Combined Myocardium and Cavity Dice Score, Cavity Dice Score, Myocardium Dice Score, Hausdorff (HD) Distance, difference between predicted and ground truth EF values and ground truth EF values.

The downside of using EF GT as a classification parameter only becomes apparent when operating at a low FPR. If, for instance, the objective was to minimise false positive cases, then using EF GT as a parameter would result in a much lower chance of picking up true positive cases than for example with using Dice scores. However, in clinical practise, this scenario usually does not occur, therefore it is of negligible concern.

6.4.3 Classification CNN with ED & ES Ground Truth Data

CNN Structure

Rather than using a direct linear classification strategy based on calculated metrics, the strategy here is centered around training a CNN to perform MACE classification. This MACE classifier network has the structure of a standard 3D convolutional classification network whereby blocks of convolutional, batch normalisation, leaky ReLU and max-pooling layers are followed by a fully-connected Dense layer and a softmax activation for a 2-class prediction of outputs into MACE and non-MACE outcomes. The structure of the network can be found in Figure 6.8. Due to time constraints, no systematic hyperparameter optimisation was performed. Instead, the given architecture was chosen based on previous experience, relevant literature and computational constraints. Binary cross-entropy formed the network's loss function and an Adam optimizer was used with a learning rate of 10^{-4}.

Training Protocol

Due to having to use MACE data also for training in this case, a very low number of cases was left for testing. Hence a nested cross-validation approach was used with 5×5 folds. Similarly to the generative network training in Part I non-MACE ED/ES data was split into train/validation/test folds with the same $\sim 7/1/2$ ratio, whilst also applying online augmentation of rigid rotations and translations to training and validation data to yield 6800 and 380 non-MACE cases respectively. Non-MACE testing data comprised 188 cases per fold which accumulated to a total of 940 test data pairs overall.

MACE data was equally split along the $\sim 7/1/2$ ratio (train=50, validation=8, test=15) and augmentation was applied to increase training and validation data sets to yield the same numbers as for the non-MACE set. However, as MACE test data only contained 15 cases, the nested cross-validation approach was used. Therefore for each fold of non-MACE data, 5 separate training cycles were conducted, each with a different fold of MACE data. Furthermore, during testing MACE test cases were

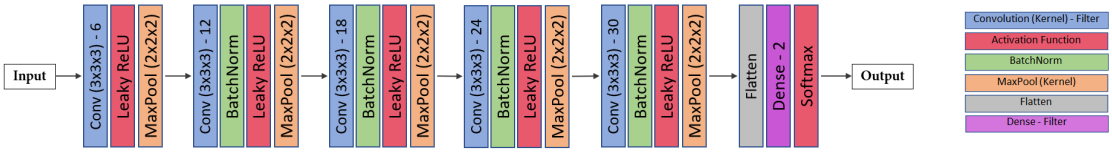


Figure 6.8: MACE Classification Network structure.

repeated to create a total number equal to the number of non-MACE test cases in order for the overall prediction accuracy to not be skewed by class imbalance. Results from all nested folds were then averaged for the final prediction result. These were subsequently used to generate ROC curves and calculate corresponding AUC results.

Network Variants

Given the relative success of the EF GT as a classification parameter in the direct linear classification stage, as well as its common use in clinical practice, the idea was to augment the MACE classification network architecture to incorporate additional EF information. Thereby, it was argued, the network could be forced to place more value on implicitly computing the ejection fraction with the hope that this would improve classification performance.

Incorporating EF information into the network was achieved through two different methods, as outlined in Figure 6.9. Firstly it was done by introducing EF as an extra output and regression loss, in addition to the pre-existing binary cross-entropy loss for MACE classification. The final fully-connected Dense layer hence contained three instead of two output values. The additional EF output then passes through a linear activation to perform a regression of the EF value for the ED and ES input volumes. The regression error was computed as a Mean-Squared Error (MSE). This additional loss was then added to the BCE loss with a relative weighting of 0.1 to form the overall network loss. Due to time constraints, optimisation of the loss weighting hyperparameter could not be conducted.

The second method constituted adding the EF value as an additional input to the network. The EF input hereby got turned into a 3D volume where each voxel's value matched the EF value. This volume was then concatenated to the ED &

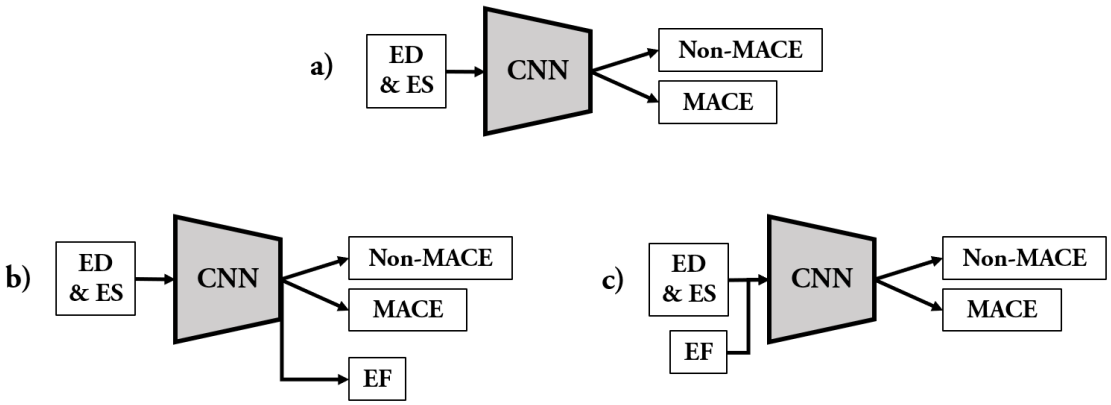


Figure 6.9: MACE Classification Network Variants. a) Standard structure with no additional EF information. b) Structure with additional EF regression loss. c) Structure with Additional EF input

Table 6.5: AUC Scores of MACE Classification Network Variants.

	MACE Classification Network Variants		
	No EF addition	**Additional EF loss**	**Additional EF input**
AUC Scores	0.733	0.731	0.748

ES volumes prior to the first convolutional layer. Note that this network variant did not use the previously described EF regression output, as it was of interest to separate the methods to understand their respective effect on network performance.

Experiments

All three networks variants, i.e. using no EF information, using EF as an additional loss and using EF as an additional input, were trained for up to 100 epochs whilst employing early stopping triggers based on validation loss increases, as mentioned previously. All networks reached optimum training before the final epoch, as indicated by overfitting in the training curves.

Results

The AUC scores for each of the three MACE Classification Network variants were computed based on calculated ROC curves and can be found in table 6.5.

Discussion

Firstly, based on results in table 6.5 it can be seen that by training a CNN on full ED and ES volumes of non-MACE and MACE subjects, a classification AUC of 0.733 can be achieved which significantly outperforms a direct linear classification using EF GT alone (AUC of 0.674). This demonstrates that additional discriminative information is contained in the LV shape which cannot be captured by an EF score alone. This additional information is highly beneficial for predicting MACE risk.

Secondly, the two additional variants have shown that incorporating EF information into a classification CNN can help improve classification accuracy, however, the method of incorporation matters. Adding EF information through an additional regression loss has shown to confuse the network more rather than help steer it towards better classification, though the difference in performance to the non-EF variant is negligible. Adding EF as an additional input, however, seems to help boost the performance by a non-negligible amount. The input method for metadata matters, as previously seen in chapter 4 where it was shown that adding data to the first layer of the network is most beneficial.

MACE risk prediction based on the same data set has been attempted through other methods such as global long axis strain (GLS) [281] and MAPSE [282] where AUC scores of 0.748 and 0.74 were achieved respectively. Further methods using multivariable shape analysis including Killip class [280] have achieved AUC scores of 0.747. The method presented here has therefore achieved values similar to the state-of-the-art.

6.4.4 Classification CNN with ES GT & Prediction Data

In contrast to the previous network setup, the idea in this section was not to use ground truth ED & ES volumes as the network input, but instead use the difference between ES ground truths and their associated ES predictions, as outputted by the 3D UNET in Part I. The reason for doing so was twofold.

Focus on Function

Firstly, the drawback of using ground truth ED & ES volumes as inputs is that the network only sees structure and function relating to a specific input case, yet without the context of how this case relates to "normal" non-MACE function. Using the difference between ES ground truth and ES prediction, however, means that the input already contains information about how the function of a specific input case relates to and differs from "normal" non-MACE function. Therefore during classification the network is able to focus more directly on the difference in function between this specific case and the wider non-MACE population, rather than learning a broader distribution of both structure and function.

On top of that, in the case of ED & ES inputs, the network additionally receives structural information which makes downstream disentanglement of structural and functional features more difficult. As it is the aim of this section to understand how non-MACE and MACE function differ, this is important.

Visualisation of Pathophysiology using CAMs

The second reason is related to the understanding and explainability of the classification mechanism and how this helps with learning about MACE-related pathophysiology. Neural networks have long been treated as "black boxes" due to the difficulty of understanding their inner workings. However, methods such as class activation maps (CAMs) can help alleviate such issues. CAMs are employed to better interpret and understand the class prediction made by a network. They highlight the discriminative regions in the input used by the classifier in its decision making process. To do so, the activations of the final convolutional layer are taken and up-sampled to the original input dimensions. They can then be overlaid onto the input to highlight regions of interest for the classifier. Here the choice of using a single difference volume, rather than two separate volumes for the input becomes more apparent, as it is easier to up-sample to the former rather than the latter. Creating CAMs and combining them with 3D renderings of input volumes can give a better understanding as to which volumetric regions in the

LV differences between ES predictions and ES ground truths are indicative of a non-MACE vs. MACE classification.

Experiments

To test both the classification performance of the CNN using differences between ES ground truth and prediction volumes, as well as implement CAMs to visualise regions of interest related to MACE pathophysiology, the CNN from the previous section was re-trained. All parameters, except the input, remained the same. The AUC was calculated based on the computed ROC for the test sets. In terms of data, the 940 non-MACE and 240 MACE test outputs from Part I were used and split into training, validation and test data sets according to the $\sim 7/1/2$ ratio from before. Rigid rotational and translational augmentation was used for training and validation data sets, leading to 6800 and 350 non-MACE and MACE cases each for training and validation respectively. Five fold cross validation was applied and each fold consisted of 180 non-MACE and 48 MACE test cases. The latter were repeated to reach 180 cases in total in order to address class imbalance. Prediction and activation results across all folds were averaged.

Results

An AUC of 0.723 was achieved by the network. Figure 6.10 shows 2D short-axis and long-axis slices of the average ES ground truth test set myocardium along the z-and x-axis with corresponding overlaid CAMs. These are shown in the form of heat maps, according to the scale on the right hand side of the figure, where a higher number indicates a stronger focus of the network on the indicated region. Figure 6.11 shows the same CAMs but this time overlaid onto a 3D representation of the average ES ground truth test set myocardium. Similarly, the scale on the right hand side of the figure indicates the level of attention given to specific volumetric regions. The LV myocardium is shown from all four viewpoints on the x-y grid from both the side and top to allow for a visualisation of both the endo-and epicardial surfaces.

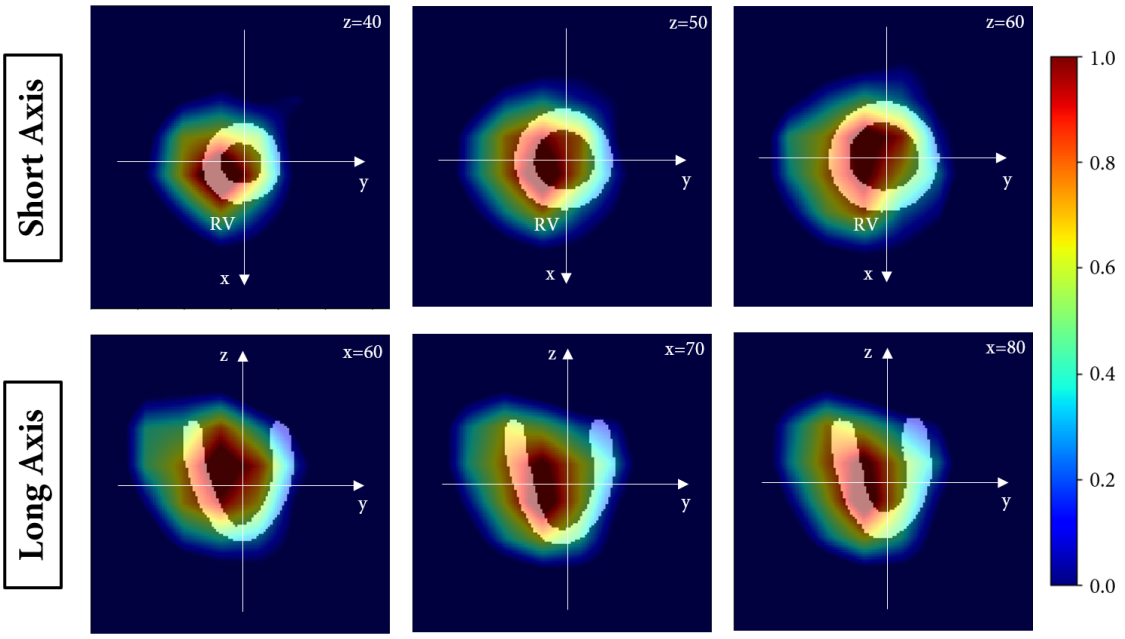


Figure 6.10: 2D short-axis and long-axis slices of the average test set ES myocardium segmentation volumes at different positions along the z-and x-axis respectively are shown together with associated class activation maps (CAMs). These highlight the activated regions in the last convolutional layer of the MACE classification network and show which parts of the input data the network is focusing on when making classification decisions. For orientation the respective position of the RV is indicated on the short-axis slices. For the long-axis slices the RV is positioned out of the page relative to the 2D slice.

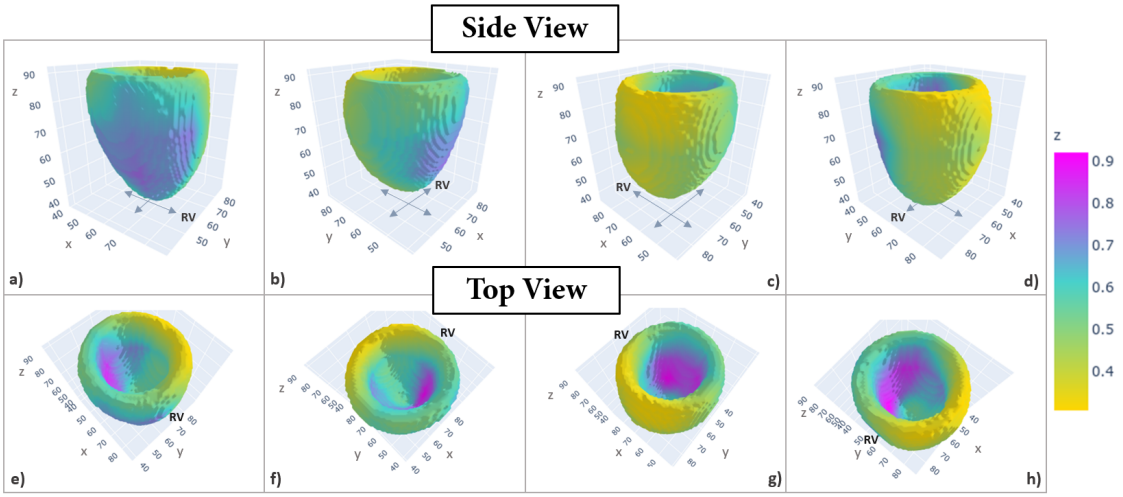


Figure 6.11: Activation maps from 6.10 rendered onto a 3D ES myocardium volume and shown from all four viewpoints on the x-y grid, both from the side and the top. The colours as given by the colour bar indicate the levels of activation. The location of the RV is also labelled.

Discussion

While the achieved AUC is lower than for any of the classifier variants using ED and ES volumes as inputs, it is still significantly higher than using EF GT or any of the other computed metrics from Part I. Furthermore, this method allows for the visualisation of regions of interest for the classifier relating to functional differences between non-MACE and MACE populations. As seen in Figures 6.10 and 6.11, the region most highlighted is the anterior wall of the LV. Figure 6.10 in particular shows that it is more specifically the endocardial side of the anterior wall that is of interest, meaning that differences in the contraction in this region are the main determinant for classification of hearts into non-MACE and MACE outcome groups. Figure 6.11 further highlights the regional spreading of the activation heat map which indicates that most of the activation is located between the apical anterior and mid anterior part of the wall, if classifying regions according to the 17-segment model. Figure 6.10 also shows slight activation near the basal anterolateral segment, again mainly on the endocardial side.

When looking back at Figure 6.6 which depicts the average ES predictions of the 3D UNET for non-MACE and MACE populations, it can be seen that there is some agreement. The regions of highest variation for MACE predictions were predominately located around the apex and the regions between apex and mid-axial locations of the heart. It does seem though that apical activation is lower than the deviations in Figure 6.6 might suggest. Equally, the deviations of predictions at the base of the heart do not seem to matter as strongly in the activation maps. This is an example of how human analysis might come to different and potentially misleading conclusion compared to neural networks.

The most activated region in the apical anterior wall on the endocardial side corresponds with an overprediction of contraction in MACE cases compared to their ground truths, according to Figure 6.6. Hence in conclusion it seems that MACE cases display a particularly weak endocardial contraction in that region compared to non-MACE cases.

6.4.5 Part II Discussion

Classification Performance

Out of the three strategies that were investigated for assessing MACE risk a classification CNN that uses both ground truth ED & ES volumes, as well as EF information as inputs is best suited for MACE outcome classification. It significantly outperforms the direct linear classification approach of using just EF ground truth values as the decisive parameter. Also in terms of absolute risk prediction accuracy, it performed better than using ES predictions from Part I.

Since AUC can also be taken as the probability that a network will classify a randomly chosen positive instance as higher than a randomly chosen negative one, it can be said that a MACE outcome classification accuracy of almost 75% was achieved. In a wider context this makes this network capable of competing with other state-of-the-art methods for MACE outcome prediction.

Pathophysiology Visualisation

Whilst it would be too far-fetched to draw medical conclusions based on the activations in Figures 6.10 & 6.11, the highlighted regions may be used to confirm pre-existing understanding of MACE pathophysiology and help validate the use of the methods in this chapter. As mentioned in the introduction, however, given the generic nature of the term "MACE" which encompasses a wide range of clinical outcomes, derived physiological understanding from the activation maps cannot be directly linked to specific outcomes. Nonetheless the location of activations is to some degree validated by [280] who identified a similar impairment of the anterior wall as a major contributor towards MACE risk. This area is known to be a predominant region of remodelling following MI, whereby the posterior wall seems to pull the weakened anterior wall, suggesting a perfusion impairment in the left anterior descending artery, in agreement with [283]. So whilst no definitive clinical conclusion can be drawn, this does seem to suggest the clinical potential of activation maps such as these. Generally speaking MACE is very broad term which could be sub-divided into smaller outcome categories, however, this would

also mean decreasing the already quite limited data size which would affect network performance. Hence at the design of the study it was decided that MACE would be the best endpoint, a convention that was adopted in this chapter. In future work, more nuanced activations which also relate to micro-structural features would be a useful. Alternatively it could be of interest to examine other frames and model the difference of activations along the sequence.

In addition to activation maps as a method of interpretability for DL systems other methods such as Testing Concept Activation Vectors (TCAV) could be applied. These have been used by Jannik et al. [284] for trying to discover concepts representative for different cardiac conditions from the ACDC challenge data set [201]. TCAV [285] aim to show how much a human interpretable concept is useful for a prediction in a pre-trained model, even if the concept was not included during training. It does so by finding concept activation vectors (CAV) which are the normals to a hyper plane in the network layers' activations separating examples with and without a specific concept. Concept in the medical context could be the presence, shape or size of particular anatomical structures.

Limitations & Improvements

Limitations are strongly related to the data set that is used. Apart from the moderate size, assumptions were made related to the acuteness of MIs and the time span after which patients were imaged. Furthermore, in the 12 month period after which a re-assessment of outcomes was taken, several patients had suffered from congestive heart failure (n=20), re-infarction (n=21) or death (n=32). However, no information was available as to whether patients had fallen into several categories, or whether death was necessarily due to the initial diagnosis of myocardial infarction or other causes, leading to an overall level of uncertainty in terms of possible correlations or even causalities that could be made between structural changes and outcomes. The imbalance of non-MACE and MACE data also created difficulties that needed to be addressed. Lastly, the process of segmentation, mesh creation and segmentation volume estimation adds uncertainties at each step. Working

directly on mesh data, using graph or point cloud based network versions rather than voxel grids would likely help alleviate this issue. CAMs as implemented in this chapter are useful at highlighting regions of interest. However, they provide blurry and insufficiently localised activations, due to the upsampling steps inherent in the method. Improvements on the CAM method as released in the GradCAM++ paper [286] could form potential alternatives in the future. Using voxel grids to represent the data means that the inherent sparseness in the data cannot be used to its full advantage. Training processes are slow and computationally expensive. Alternative data representations in the form of meshes and graphs could be suitable alternatives here.

Classification performance could have been improved through a number of methods. Firstly, no hyperparameter optimisation was performed for the MACE classifier. Though it is thought to be unlikely that it would be a major factor for improving accuracy, it would probably have had a positive impact. Furthermore, as with most deep learning networks, the classifier would benefit from a larger data set. Augmentation is a useful tool for achieving this to some degree, but it does not replace real data. Increased data set size would allow for network capacity to increase without running the risk of premature overfitting. To allow for larger networks to train, an increase in computational resources would also be necessary. More computational power would equally make it possible to use larger batch sizes which had to be substantially reduced in order to accommodate for the memory requirements of 3D data. Larger batch sizes help with training stability and speed. This could also be achieved by utilising pre-trained networks, either from other binary 3D applications such as Vox Net [75] or by having the data run through an autoencoder first from which the encoder is then taken as the classifier and fine-tuned. With regards to the EF metadata that forms the additional input, using braided nets [260], as previously mentioned in chapter 4, could provide better metadata integration. Further metadata could also be integrated, such as ECG or strain data which has proven highly indicative of MACE outcome [281, 282].

6.5 Conclusion

This chapter has shown that 3D cardiac function can be modelled using volume-to-volume translation networks, such as the 3D UNET. More specifically, it was demonstrated that by training such a network on a sub-population of subjects, namely MI patients who did not subsequently experience MACE, the network was able to extract population specific function information that could not be found in MI patients that did experience MACE. The difference in performance of those groups proved to be statistically significant. On a qualitative level, the 3D UNET enabled an insight into the average regions of mispredictions which provides an idea of the functional differences that exist between the sub-populations.

The chapter has also demonstrated that whilst EF is a better indicator of MACE outcomes when compared to other derived functional metrics, using deep convolutional neural networks is a substantially more accurate way of predicting MACE outcomes. An increase from 67.4% to 74.8% in prediction accuracy could be observed which constitutes a relative improvement of 11%. The best architecture was one that used both ED & ES volumes as its input in addition to a pre-calculated EF value. This was able to achieve state-of-the-art MACE outcome prediction performance. Even better prediction results could likely have been achieved, however, a full optimisation of the method is beyond the scope of this work.

When using the difference between ES predictions from the 3D UNET and their corresponding ES ground truths as classification network inputs, the implementation of class activation maps facilitated an insight into the internal reasoning of the classification network and the regions in the input volume the network gives attention to when making its decision. This provided a basis for learning more about MACE-related pathophysiology. Specifically it could be seen that the endocardial contraction of the anterior wall of the LV plays a relatively more important role for classifying MACE outcomes compared to any other LV region. While this chapter is not intended to change the medical understanding of MACE, the work aims to validate the use of deep learning methods in this area to drive future clinical integration.

Overall improvements to the methods in this chapter could include adapting the workflow to train the 3D UNET and MACE classifier together, so that the 3D UNET gets optimised towards generating a larger divergence between non-MACE and MACE outputs. This might help with MACE outcome classification accuracy when using the ES ground truth and prediction difference as the input and also result in more specific pathology related insights. A limitation with CAMs though, particularly as used in this context, is that by using a fairly deep network, the upsampling procedure that needs to be employed to transfer the activation maps back to the original dimension is quite extreme, causing blurring and reducing activation specificity.

Whilst ED and EF are thought to hold a large deal of discriminative information regarding cardiac function and are commonly used for clinical purposes, thorough function modelling would also need to include additional frames in the sequence. This would likely improve the model's predictive capabilities.

Further analysis could also focus more specifically on MACE patient subgroups. For example, it has been shown that LVEF often remains constant or only minimally decreases in patients following MIs [287], meaning that these patients only have a low theoretical risk of developing MACE. Clinically this increases the chance of a false negative if only using LVEF as the indicator. Therefore a more in depth analysis of patients with normal LVEF but high MACE risk probability could be of interest.

The key contributions of this chapter are:

- Comparative evaluation of 3D UNET, 3D cGAN and 3D cVAE architectures in their suitability for performing volume-to-volume translations to model 3D cardiac function by predicting ES segmentation volumes based on ES segmentation inputs.

- Development of an optimised 3D UNET for modelling cardiac function of non-MACE subjects to allow for extraction of MACE and non-MACE sub-population specific cardiac function information. Associated visualisation

of 3D shape information was generated to allow for the understanding of physiological differences in 3D cardiac contractions between sub-populations.

- Comparative evaluation of the suitability of clinical and function-derived metrics for MACE outcome prediction.
- Comparative evaluation of CNN-based classification methods for MACE outcome prediction.
- Development of CNN-based classification method for MACE outcome prediction with state-of-the-art accuracy.
- Extraction of MACE sub-population specific 3D cardiac function information with associated visualisation using class activation maps for physiological understanding of sub-population differences in 3D cardiac contractions.

7

Conclusions & Future Work

Contents

7.1 Summary of Results & Contributions 233
7.1.1 Capturing Cardiac Structure 233
7.1.2 Modelling Cardiac Function in 2D and 3D 234
7.1.3 Enabling a 2D to 3D Volume Reconstruction Pipeline . 235
7.1.4 Prediction of Clinical Outcomes 236
7.2 Future Work 237
7.2.1 Overcoming GAN & VAE Limitations for Improved Cardiac Structure Captioning 237
7.2.2 Personalised Function Modelling 238
7.2.3 Methodology Improvements 239
7.2.4 Drive towards Clinical Implementation 239

The overarching aim of this research was to investigate the use of deep generative networks for gaining a more comprehensive understanding of structure-function interactions in the heart, given a variety of CMR data sets from healthy and diseased patients, and to extract sub-population specific understanding of cardiac function and pathophysiology in order to generate a basis for disease phenotype and clinical outcome classification. The thesis presented a number of solutions for meeting this aim, particularly focusing on the specific goals stated in chapter 1.

This chapter summarises the results and key contributions of this thesis and explores to what extent the proposed goals have thereby been achieved. It then explores the limitations of this research and potential directions for future work to address such.

7.1 Summary of Results & Contributions

7.1.1 Capturing Cardiac Structure

It was shown that capturing cardiac structure can be approximated by learning the underlying data distribution of CMR data sets through the application of deep generative methods like GANs and VAEs to implicitly and explicitly learn distribution parameters, thereby allowing for the generation of new data instances.

To develop such GANs and VAEs chapter 3 outlined hyperparameter optimisations for both 2D GAN and VAE architectures. The relative performance of network variants was evaluated using Frechet Inception Distance (FID) and qualitative rating and classification scores. "Optimum" GAN and VAE networks were developed with respective FID scores of 996 and 1700 and qualitative rating scores of 4.2/5 and 3.4/5. Furthermore, real and fake (i.e. generated) sample images were classified into their respective categories with a 65% and 97% accuracy, where an accuracy of 50% shows perfect assessor confusion and therefore optimum network performance. Overall this highlighted that GANs are strongly capable of capturing cardiac structure with a high degree of "realness" in image appearance that matched the performance of similar work in [146]. However, GANs have been shown to struggle with achieving accurate cardiac geometry. Auto-encoder pre-training of the generator has not

helped to reduce this. Whilst VAE sample geometry is better than those from GANs, VAEs labour with problems of blurriness, therefore samples are often detected as fake. Nonetheless, VAE latent spaces have shown to be able to extract modes of variation from within CMR data sets which correlate which anatomical and image-based features. Furthermore, by manipulating the latent space, the generation of synthetic image data can be steered into over or under expressing specific modes, thereby generating instances belonging to specific patient sub-populations.

7.1.2 Modelling Cardiac Function in 2D and 3D

Having learnt to capture cardiac structure, modelling cardiac function conditioned on structural inputs formed the next challenge. To address this, the function modelling process was condensed into predicting 2D end-diastolic frames from end-diastolic inputs using a conditional GAN, thereby capturing the main aspect of cardiac function by focusing on the extreme points in the cardiac sequence.

This method has been shown to achieve a high frame prediction accuracy, with mean MSE and SSIM scores of 0.0026 and 0.89 respectively and mean myocardium Dice scores following ES frame segmentation of 0.89.

Furthermore, this network showed that by training on sub-populations of subject data, cardiac function can be extracted that is specific to a sub-population and cannot be found in other groups. Statistically significant differences in prediction accuracy that were found between healthy UK Biobank data sub-populations differentiated by gender, age and BMI and diseased sub-populations differentiated by a diagnosis of hypertrophic cardiomyopathy support this argument. Qualitative differences in function could also be shown for these groups, validating the method for its future use in clinical settings.

This process was repeated in 3D, using ED and ES segmentation volumes of myocardial infarction survivors in a 3D UNET architecture, where it showed mean prediction scores of 0.915, 0.858 and 0.795 for epicardium, cavity and myocardium Dice scores. Furthermore a mean Hausdorff Distance of 7.24 and an absolute difference in ejection fraction of 0.082 were achieved. Equally this network showed

statistical significance for the difference in results of two sub-populations, MACE and non-MACE, when trained on the latter. The qualitative insights into MACE pathophysiology that were gained from this can be used to validate the method for clinical settings.

To facilitate the scale-up of sub-population specific function modelling, subject metadata was included into the 2D cGAN architecture. This improved ES prediction accuracy, as measured by MSE score, by an average of 4.8% across all meta architectures with a maximum of 5.7% for the best performing architecture when training and testing on a mixed data set. Respectively, mean and maximum SSIM, cavity Dice and myocardium Dice scores changed by 0.16% & 0.3%, -0.16% & -0.3% and 0.6% & 0.7%.

When testing on a specific sub-population of data, however, chapter 4 has shown that training on a mixed data set including metadata does not significantly outperform training directly on an equally sized sub-population of data. Generally though it can be said that using metadata in cardiac function modelling has been shown to be effective and promising in improving patient specific predictions and extracting specific physiology information.

Overall it can be summarized that whilst numerous methods exist for modelling cardiac function using different approaches such as velocity and deformation fields, strain curves, and complex atlas-based methods, using cGANs with additional patient metadata has been shown to offer a potentially simpler and more holistic approach.

7.1.3 Enabling a 2D to 3D Volume Reconstruction Pipeline

Current and future pipelines for reconstructing 3D CMR volumes from 2D slices will utilise conditional generative networks given 2D segmentation inputs. The more slice orientations are available, the higher the reconstruction accuracy will be. Particularly for the UK Biobank several applications using 3D data are envisioned. Given the lack of publicly available segmentation networks for UK Biobank 2-chamber CMR slices, and manual segmentations being extremely time consuming,

an automatic segmentation network was developed for this purpose, based on a proprietary cGAN architecture, as seen in chapter 5.

Using this network, epicardium, cavity and myocardium Dice scores of 0.972, 0.962 and 0.906 respectively were achieved. Additionally, the fraction of test set cases that exhibited a fracture in the myocardium segmentation label could be kept to $< 5\%$ of the total test set using this network.

7.1.4 Prediction of Clinical Outcomes

Learning structure-function interactions in the heart ultimately serves the purpose of providing a more thorough understanding of cardiac pathologies and thereby facilitates more accurate and patient-specific clinical outcome prediction. This is one of the central aims of this research. Clinical outcome prediction can be done directly through linear classification based on clinical parameters related to global cardiac function. This approach has its limitations though and is widely deemed as insufficient for future clinical practise [2, 3]. Using deep classification CNNs, as outlined in this research, can provide superior outcome prediction results. As shown in chapter 6, MACE risk prediction can be achieved with an accuracy of 74.8% which forms the state-of-the-art on this data set.

However, straightforward classification networks often lack explainability, suffering from the "black box" problem of deep learning. Using information derived from 3D functional modelling with MACE and non-MACE patients, this problem was overcome. By applying class activation maps, 3D functional differences between MACE and non-MACE patients which classification networks deemed as indicative of MACE outcome risk could be highlighted. Specifically it was shown that anterior LV wall contractions, particularly on the endocardial side, form functional regions of interest for MACE classification.

The work relating to explainable outcome prediction, as shown in chapter 6, lays the groundwork for future clinical implementation and patient-specific cardiac analysis.

7.2 Future Work

7.2.1 Overcoming GAN & VAE Limitations for Improved Cardiac Structure Captioning

Many of the limitations of the research in chapter 3 regarding the accurate capture of cardiac structure are strongly related to the inherent weaknesses of GANs and VAEs, namely the inability to consistently produce accurate cardiac geometries (in GANs) and the problem of image blurriness in VAEs. To overcome these limitations, a VAE-GAN [133] could be implemented instead. VAE-GANs utilize the VAE structure, yet with an additional GAN discriminator attached to the output. This judges the VAE outputs in terms of their "realness" by employing an adversarial loss which then gets added to the total network loss. VAE-GANs have shown to overcome the problem of blurriness better than traditional VAEs [172], but can also tap into a regularized latent space for stronger output geometry control. In terms of validating outputs, similar to [146], the generated VAE-GAN output could be further validated by testing it against real data in training a segmentation network. This could provide a promising avenue of future research. Alternatively architectures such as Adversarial Autoencoders [169] could also be explored.

Irregardless of the global architecture used, more detailed changes such as using deconvolutions, otherwise known as transposed convolutions, instead of the current approach of up-sampling layers + convolutions could be implemented. It has been shown that using the former yields better results.

With regards to synthetic data set generation applications, particularly those that are guided towards specific data sub-populations by manipulating the latent space, a better disentanglement of the latent space is necessary. As seen in chapter 3, latent space vector elements correspond with anatomically identifiable changes in the output, e.g. increases in LV cavity size or changes in SAX location towards apex or base, however, these are often highly compounded. InfoVAEs [182] which add maximum mean discrepancy to the latent to further regularise it could offer an interesting extension.

Lastly the Kaggle Competition data set is extremely heterogeneous in its composition. For applications where specific output data is required, for example mid-axial SAX slices at end-diastole, having the Kaggle data set as a training data source could mean that produced output data might require post-processing to filter out images that do not satisfy the required criteria. Training on data sets like the UK Biobank might be more suitable, as it is much more thoroughly labelled and has been acquired using the same protocol across scanners, thereby ensuring much higher data set homogeneity.

7.2.2 Personalised Function Modelling

Functional modelling, as shown in chapter 6, can be utilised for outcome classification purposes whilst providing clinical explainability. To improve such processes further personalisation of cardiac modelling is crucial.

Including metadata into the modelling process has shown to be a promising avenue, though the method is not optimised yet. Braided blocks [260] have shown to be useful in adapting metadata to the feature domain and increase its predictive value. These should therefore be explored. Additional patient metadata should also be included. This is readily available in the UK Biobank and includes patient characteristics, patient and family history and genetic data. Including these is of particular interest in for 3D modelling performed in chapter 6, akin to metadata inclusion experiments in chapter 4.

Multi-modality models could combine CMR data with other sources of relevant data to generate more comprehensive approaches for function modelling and disease prediction. Such sources could include biomechanical data such as myocardial strain information which can be gathered through MR Tagging or Speckle Tracking Echocardiography, or time-series data such as ECG. These sources provide regional and global information that is not generally picked up on CMR, yet holds rich diagnostic values [281, 282].

7.2.3 Methodology Improvements

Whilst 3D cardiac function modelling is likely to be the way forward, due to the richness of 3D volumetric data, dealing with dense voxel grid data requires significant memory. Point Cloud Network approaches, as outlined in A.1, could be explored further for this purpose. Similarly, working directly on mesh data using graph neural networks (GNNs) [288] is a promising avenue. These offer much higher resolution for lower memory requirements, thereby allowing 3D cardiac function to be modelled with more detail.

Furthermore, the use of attention blocks should be considered. Transformer based networks have become well established in the image analysis field and have shown strong success in guiding classification network's attention to relevant regions [61, 62]. This could significantly improve MACE outcome prediction.

Lastly, expanding the methodology to the full cardiac sequence should be considered, as this will allow for information sitting outside the ED & ES frames to be picked up. HCM for example is a pathology that predominately affects the dilation process. Therefore, examining frames sitting between ES and ED could yield valuable insights into the disease and its heterogeneous phenotype nature.

7.2.4 Drive towards Clinical Implementation

The research presented in this work served both the purposes of validating the use of generative methods in capturing cardiac structure and modelling corresponding cardiac function, as well as demonstrating the use of such methods in clinically relevant applications. Clinical implementation of methods such as those presented in this thesis is absolutely crucial for making an actual impact on patient care. Yet for getting to that stage several hurdles have to be overcome. One of the major hindrances is the issue of trust. How can clinicians and health care policy makers trust DL systems and therefore ensure their safety and benefit for patients? Validation is key in this regard. In this case the classification process outlined in chapter 6, including the integrated explainability of network workings, could be applied to a range of additional outcome data. Hereby the UK Biobank provides

a rich source of data with a wide range of pathological cardiovascular outcomes. Applications of the discussed methods to data sets such as these could provide additional forms of performance validation which could help with the generalisation of the methods and build confidence in their applicability and safety. Further levels of interpretability and explainability of inner network workings, using activation maps, TCAV methods or others would allow users to gain meaningful insights into and comprehension of the reasoning of DL systems and build trust. Limitations to implementation are varied but among others include the cost of re-training clinicians, the prevalent use of commonly available echocardiography machines and the sometimes significant manual interaction required for processing steps.

In the future, in order to facilitate truly personalised medicine, deep generative methods need to be developed that allow for patient metadata, family history and different data modalities to be fused together to facilitate accurate and individual modelling of cardiac structure-function interactions for each patient.

Appendices

A

Supplementary Section for Chapter 4

Contents

A.1 Point Cloud Network Extension 242
A.1.1 Disclaimer 242
A.1.2 Introduction 243
A.1.3 Methods 243
A.1.4 Results 243
A.1.5 Conclusion 245

A.1 Point Cloud Network Extension

A.1.1 Disclaimer

As part of a collaborative project it was decided to extend work of the chapter 4 section "cGAN for the Prediction of Cardiac Contraction" to 3D with the use of Point Cloud Networks. This is outlined in the joint publication by Beetz, M., Ossenberg-Engels, J. et al.: Marcel Beetz, Julius Ossenberg-Engels, Abhirup Banerjee, and Vicente Grau, "*Predicting 3D Cardiac Deformations With Point Cloud Autoencoders*", International Workshop on Statistical Atlases and Computational Models of the Heart (STACOM) (Accepted), Strasbourg, France, pp. 1–8, September 2021

A.1.2 Introduction

This work builds on a previous paper by Beetz et al. [289] which develops a pipeline for reconstructing biventricular surfaces from MRI contours using Point Completion Networks [290]. Hereby, short and long-axis CMR frames were taken and segmented to extract biventricular contours. These were then joined in a 3D array to form a sparse and misaligned 3D point cloud which was subsequently fed into the Point Completion Network similar to [290] to render it into a dense point cloud with active misalignment correction. Lastly this cloud was converted into a mesh using the common ball pivot algorithm.

A.1.3 Methods

In [289], this pipeline was first used to create a data set of dense and misalignment corrected point clouds for pairs of ED and ES frames from the UK Biobank. A total of 1000 ED and ES point cloud pairs from 500 male and 500 female subjects was thereby generated. Hereby point clouds contain the same number of points for both frames, though points between frames are not registered to one another. The Chamfer distance which is used in this application as a metric of accuracy measures the distance between nearest points. It is hereby not linked to any two specific points.

Akin to the work in chapter 4 of this thesis, the paper then tries to predict ES point clouds based on ED point cloud inputs, thereby modelling biventricular function in a simplistic way. It does so using a network that forms an extension of the PointNet++ [291] architecture. This network has been adapted in conjunction with [289] and [290] to build a 3D Point Cloud Network which works as a point cloud to point cloud transfer network, similar to a UNET. The idea was to repeat the ED to ES translation work from this chapter using said network.

A.1.4 Results

It was shown that the PCN network is very capable of achieving this task. Given its nature as a point cloud network, performance evaluation is not based on image metrics such as MSE or SSIM, but instead utilized the Chamfer Distance to measure

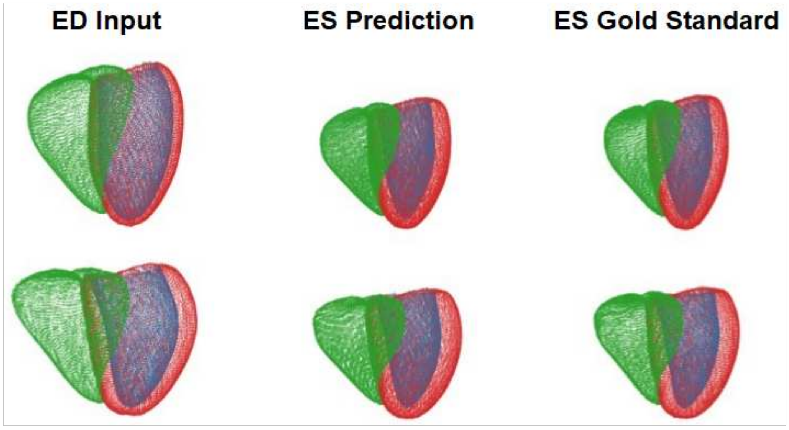


Figure A.1: ES prediction results for two sample cases of the unseen test data set. Figure is courtesy of Marcel Beetz.

distances between surfaces. Experiments included both an ED to ES translation, as well as an ES to ED translation. It was shown that average Chamfer Distances for the predictions lay close to 1.88mm which is the pixel size in the ground truth. Results with a higher level of accuracy than the inherent inaccuracy of partial volume effects of the ground truth are not achievable, thereby indicating high performance by the point cloud network. Based on the predictions, clinical metrics such as myocardial mass, ventricular volume, ejection fraction and stroke volume were calculated which scored within 5% of their ground truth values. Lastly, a network was trained on ED and ES data pairs stemming from just female subjects, and subsequently tested on both female and male subject data. Similar to the work presented earlier in this chapter and published in [244] it was shown that the network is able to produce statistically significantly better predictions when tested on data with similar population characteristics (i.e. also female) than different ones (i.e. male). This indicates that certain deformation patterns exist in hearts of the female sub-population that cannot be found in males. The paper extends this by also going in the opposite phase direction, predicting ED based on ES inputs and shows that this equally produces statistically significant differences between females and males. The KS-test was used to test for statistical significance. Whilst point to point registration did not exist, problems with varying matching of points between clouds was overcome by pre-aligning point clouds in their 3D reference frame.

A.1.5 Conclusion

Despite the similarities mentioned, there are several differences of note between this joint work and the 2D work shown in this chapter based on [244]. Since the PCN approach works with sparse point cloud data, rather than dense, grid-based image data, it is more easily scalable to deeper networks and lends itself well to an expansion to 3D. Through this it is capable of modelling biventricular deformation in a more comprehensive way. Furthermore, it does not require any additional downstream segmentation in order to derive clinical metrics from the data, making it more easily applicable. It does, however, require extensive pre-processing steps to obtain the point cloud data from the initial CMR images.

References

[1] WHO. *Cardiovascular Disease Death Rate*. 2019. URL: https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds) (visited on 07/13/2021).

[2] Suzanne de Waha et al. "Impact of early vs. late microvascular obstruction assessed by magnetic resonance imaging on long-term outcome after ST-elevation myocardial infarction: a comparison with traditional prognostic markers". In: *European Heart Journal* 31 (21 Nov. 2010), pp. 2660–2668. URL: https://pubmed.ncbi.nlm.nih.gov/20675660/.

[3] Otto A. Smiseth et al. "Myocardial strain imaging: how useful is it in clinical decision making?" In: *European Heart Journal* 37 (15 Apr. 2016), pp. 1196–1207. URL: https://pubmed.ncbi.nlm.nih.gov/26508168/.

[4] Ian J. Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 27* (2014). Ed. by Z. Ghahramani et al., pp. 2672–2680. URL: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[5] J. Gordon Betts et al. *Heart Anatomy - Anatomy and Physiology*. OpenStax, Apr. 2013. URL: https://openstax.org/books/anatomy-and-physiology/pages/19-1-heart-anatomy.

[6] Boston University School of Public Health Wayne W. LaMorte. *Cardiovascular System*. URL: https://sphweb.bumc.bu.edu/otlt/mph-modules/ph/ph709_heart/ph709_heart2.html (visited on 08/04/2021).

[7] Biologydictionary.net Editors. *Thoracic Cavity - Definition & Organs of Chest Cavity*. URL: https://biologydictionary.net/thoracic-cavity/ (visited on 08/04/2021).

[8] B. Cornell. *Heart Structure | BioNinja*. URL: https://ib.bioninja.com.au/standard-level/topic-6-human-physiology/62-the-blood-system/heart-structure.html (visited on 08/04/2021).

[9] P.A. Iaizzo. *Handbook of Cardiac Anatomy, Physiology, and Devices*. Springer International Publishing, 2015. URL: https://books.google.co.uk/books?id=4PLuCgAAQBAJ.

[10] Heart Foundation. *Coronary Artery Disease CAD- Ischaemic Heart Disease*. URL: https://www.heartfoundation.org.nz/your-heart/heart-conditions/coronary-artery-disease (visited on 08/04/2021).

[11] Rafael Costa et al. *Vascular Diseases for the Non-Specialist*. Springer International Publishing, 2017.

[12] Anthony V. Delicce et al. *Physiology, Frank Starling Law*. StatPearls Publishing, Feb. 2021. URL: https://www.ncbi.nlm.nih.gov/books/NBK470295/.

[13] Lindsay M. Biga et al. *Cardiac Cycle*. URL: https://open.oregonstate.education/aandp/chapter/19-3-cardiac-cycle/ (visited on 11/05/2021).

[14] Brian A. Houston et al. "Hypertrophic cardiomyopathy: a review." In: *Clinical Medicine Insights. Cardiology* 8 (Suppl 1 2014), pp. 53–65. URL: http://www.ncbi.nlm.nih.gov/pubmed/25657602%20http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4309724.

[15] Martin S. Maron and Barry J. Maron. "Hypertrophic cardiomyopathy". In: *The Lancet* 381.381 (2013), pp. 242–55. URL: http://dx.doi.org/10.1016/.

[16] Advanced Cardiovascular Institute. *Dilated Cardiomyopathy*. URL: https://www.advancedcardiovascular.org/services-williamsburg/common-cardiac-conditions/dilated-cardiomyopathy/ (visited on 08/04/2021).

[17] Jeffrey B. Geske et al. "Hypertrophic Cardiomyopathy". In: *JACC: Heart Failure* 6.5 (2018), pp. 364–375.

[18] Ali M. Agha et al. "Role of cardiovascular imaging for the diagnosis and prognosis of cardiac amyloidosis". In: *Open Heart* 5 (2 Nov. 2018). URL: https://openheart.bmj.com/content/5/2/e000881.

[19] Allen J. Taylor. *Atlas of Cardiovascular Computed Tomography: Expert Consult - Online and Print Imaging Companion to Braunwald's Heart Disease*. 1st Edt. Saunders, 2009.

[20] Cedars-Sinai. *Myocardial Perfusion PET Stress Test*. URL: https://www.cedars-sinai.org/programs/imaging-center/med-pros/cardiac-imaging/pet/myocardial-perfusion.html (visited on 08/08/2021).

[21] Trisha L. Roy et al. "MRI for peripheral artery disease: Introductory physics for vascular physicians". In: *Vascular Medicine (United Kingdom)* 23 (2 Apr. 2018), pp. 153–162.

[22] William Reusch. *NMR Spectroscopy*. URL: https://www2.chemistry.msu.edu/faculty/reusch/virttxtjml/spectrpy/nmr/nmr2.htm (visited on 02/02/2022).

[23] John P. Ridgway. "Cardiovascular magnetic resonance physics for clinicians: part I". In: *Journal of Cardiovascular Magnetic Resonance 2010* 12 (1 Nov. 2010), pp. 1–28. URL: https://jcmr-online.biomedcentral.com/articles/10.1186/1532-429X-12-71.

[24] Allen D. Elster. *SSFP - Questions and Answers in MRI*. URL: http://mriquestions.com/what-is-ssfp.html (visited on 02/02/2022).

[25] Gerald M. Pohost. "The History of Cardiovascular Magnetic Resonance". In: *JACC: Cardiovascular Imaging* 1 (5 Sept. 2008), pp. 672–678.

[26] Cormac Farrelly et al. "ECG-gated multiecho Dixon fat-water separation in cardiac MRI: Advantages over conventional fat-saturated imaging". In: *American Journal of Roentgenology* 199 (1 July 2012), pp. 74–83.

[27] Klaus Scheffler et al. “Principles and applications of balanced SSFP techniques”. In: *European Radiology* 13 (11 Nov. 2003), pp. 2409–2418. URL: https://pubmed.ncbi.nlm.nih.gov/12928954/.

[28] Michael A. Guttman et al. “Analysis of Cardiac Function from MR Images”. In: *IEEE Computer Graphics and Applications* 17 (1 Jan. 1997), p. 30. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2396596/.

[29] Manuel D. Cerqueira et al. “Standardized Myocardial Segmentation and Nomenclature for Tomographic Imaging of the Heart A Statement for Healthcare Professionals From the Cardiac Imaging Committee of the Council on Clinical Cardiology of the American Heart Association American Heart Association”. In: *Circulation* 105 (4), pp. 539–542. URL: http://circ.ahajournals.org/content/circulationaha/105/4/539.full.pdf.

[30] Anca Florian et al. “Cardiac Magnetic Resonance Imaging in Ischemic Heart Disease”. In: *Journal of Medicine and Life* 4 (Nov. 2011), pp. 330–45.

[31] Andrew E. Arai and Li-Yueh Hsu. “Myocardial Perfusion and Late Gadolinium Enhancement Imaging in Cardiovascular Magnetic Resonance to Assess Coronary Artery Disease”. In: *Contemporary Cardiology* (2019), pp. 185–203. URL: https://link.springer.com/chapter/10.1007/978-1-4939-8841-9_12.

[32] Frank Rosenblatt. “The Perceptron: A Probabilistic Model for the Information Storage and Organization in the Brain”. In: *Psychological Review* 65 (6), pp. 19–27.

[33] Jürgen Schmidhuber. “Deep learning in neural networks: An overview”. In: *Neural Networks* 61 (2015), pp. 85–117. URL: https://www.sciencedirect.com/science/article/pii/S0893608014002135.

[34] Alex Krizhevsky et al. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[35] Facundo Bre et al. “Prediction of wind pressure coefficients on building surfaces using artificial neural networks”. In: *Energy and Buildings* 158 (Jan. 2018), pp. 1429–1441.

[36] Ian J. Goodfellow et al. *Deep Learning*. URL: http://www.deeplearningbook.org. MIT Press, 2016.

[37] Junxi Feng et al. “Reconstruction of porous media from extremely limited information using conditional generative adversarial networks”. In: *Physical Review E* 100 (3 Sept. 2019).

[38] Yann A. LeCun et al. “Efficient BackProp”. In: *Neural Networks: Tricks of the Trade: Second Edition*. Ed. by Grégoire et al. Montavon. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 9–48. URL: https://doi.org/10.1007/978-3-642-35289-8_3.

[39] David E. Rumelhart et al. “Learning representations by back-propagating errors”. In: *Nature* 323.6088 (Oct. 1986), pp. 533–536.

[40] Haotian Shi. *An Architecture Combining Convolutional Neural Network and Support Vector Machine for Image Recognition*. URL: http://users.cecs.anu.edu.au/~Tom.Gedeon/conf/ABCs2018/paper/ABCs2018_paper_39.pdf (visited on 03/05/2022).

[41] Dominik Scherer et al. "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition". In: *Artificial Neural Networks - ICANN 2010*. Ed. by Konstantinos Diamantaras et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 92–101.

[42] Yann A. LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[43] Asifullah Khan et al. "A survey of the recent architectures of deep convolutional neural networks". In: *Artificial Intelligence Review 2020* 53 (8 Apr. 2020), pp. 5455–5516. URL: https://link.springer.com/article/10.1007/s10462-020-09825-6.

[44] Matthew D. Zeiler and Rob Fergus. "Visualizing and Understanding Convolutional Networks". In: *CoRR* abs/1311.2901 (2013). arXiv: 1311.2901. URL: http://arxiv.org/abs/1311.2901.

[45] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*. 2015.

[46] Christian Szegedy et al. "Going Deeper with Convolutions". In: *CoRR* abs/1409.4842 (2014). arXiv: 1409.4842. URL: http://arxiv.org/abs/1409.4842.

[47] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *CoRR* abs/1502.03167 (2015). arXiv: 1502.03167. URL: http://arxiv.org/abs/1502.03167.

[48] Kaiming He et al. "Deep residual learning for image recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition*. Vol. 2016-December. IEEE Computer Society, Dec. 2016, pp. 770–778.

[49] Sergey Zagoruyko and Nikos Komodakis. "Wide Residual Networks". In: *CoRR* abs/1605.07146 (2016). arXiv: 1605.07146. URL: http://arxiv.org/abs/1605.07146.

[50] Christian Szegedy et al. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning". In: *CoRR* abs/1602.07261 (2016). arXiv: 1602.07261. URL: http://arxiv.org/abs/1602.07261.

[51] Saining Xie et al. "Aggregated Residual Transformations for Deep Neural Networks". In: *CoRR* abs/1611.05431 (2016). arXiv: 1611.05431. URL: http://arxiv.org/abs/1611.05431.

[52] Forrest N. Iandola et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size". In: *CoRR* abs/1602.07360 (2016). arXiv: 1602.07360. URL: http://arxiv.org/abs/1602.07360.

[53] Gao Huang et al. “Densely Connected Convolutional Networks”. In: *CoRR* abs/1608.06993 (2016). arXiv: 1608.06993. URL: http://arxiv.org/abs/1608.06993.

[54] Lei Tai and Ming Liu. “Deep-learning in Mobile Robotics - from Perception to Control Systems: A Survey on Why and Why not”. In: *CoRR* abs/1612.07139 (2016). arXiv: 1612.07139. URL: http://arxiv.org/abs/1612.07139.

[55] Asifullah Khan et al. “A New Channel Boosted Convolution Neural Network using Transfer Learning”. In: *CoRR* abs/1804.08528 (2018). arXiv: 1804.08528. URL: http://arxiv.org/abs/1804.08528.

[56] Fei Wang et al. “Residual Attention Network for Image Classification”. In: *CoRR* abs/1704.06904 (2017). arXiv: 1704.06904. URL: http://arxiv.org/abs/1704.06904.

[57] Aqsa Saeed Qureshi and Asifullah Khan. “Adaptive transfer learning in deep neural networks: Wind power prediction using knowledge transfer from region to region and between different task domains”. In: *Computational Intelligence* 35 (4 Nov. 2019), pp. 1088–1112. URL: https://onlinelibrary.wiley.com/doi/full/10.1111/coin.12236.

[58] Jie Hu et al. “Squeeze-and-Excitation Networks”. In: *CoRR* abs/1709.01507 (2017). arXiv: 1709.01507. URL: http://arxiv.org/abs/1709.01507.

[59] Dimitrios Marmanis et al. “Semantic Segmentation of Aerial Images with an Ensemble of CNNs”. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (2016), pp. 473–480.

[60] Sehla Loussaief and Afef Abdelkrim. “Convolutional Neural Network Hyper-Parameters Optimization based on Genetic Algorithms”. In: *International Journal of Advanced Computer Science and Applications* 9 (Jan. 2018).

[61] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[62] Niki Parmar et al. “Image Transformer”. In: *CoRR* abs/1802.05751 (2018). arXiv: 1802.05751. URL: http://arxiv.org/abs/1802.05751.

[63] Thomas N. Kipf and Max Welling. “Semi-Supervised Classification with Graph Convolutional Networks”. In: *CoRR* abs/1609.02907 (2016). arXiv: 1609.02907. URL: http://arxiv.org/abs/1609.02907.

[64] Sara Sabour et al. “Dynamic Routing Between Capsules”. In: *CoRR* abs/1710.09829 (2017). arXiv: 1710.09829. URL: http://arxiv.org/abs/1710.09829.

[65] Jonathan Long et al. “Fully Convolutional Networks for Semantic Segmentation”. In: *CoRR* abs/1411.4038 (2014). arXiv: 1411.4038. URL: http://arxiv.org/abs/1411.4038.

[66] Olaf Ronneberger et al. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *CoRR* abs/1505.04597 (2015). arXiv: 1505.04597. URL: http://arxiv.org/abs/1505.04597.

[67] Jiwon Kim et al. "Accurate Image Super-Resolution Using Very Deep Convolutional Networks". In: *CoRR* abs/1511.04587 (2015). arXiv: 1511.04587. URL: http://arxiv.org/abs/1511.04587.

[68] Michal Drozdzal et al. "The Importance of Skip Connections in Biomedical Image Segmentation". In: *CoRR* abs/1608.04117 (2016). arXiv: 1608.04117. URL: http://arxiv.org/abs/1608.04117.

[69] Augustus Odena et al. "Deconvolution and Checkerboard Artifacts". In: *Distill* (2016). URL: http://distill.pub/2016/deconv-checkerboard/.

[70] Özgün Çiçek et al. "3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation". In: *CoRR* abs/1606.06650 (2016). arXiv: 1606.06650. URL: http://arxiv.org/abs/1606.06650.

[71] Fausto Milletari et al. "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation". In: *CoRR* abs/1606.04797 (2016). arXiv: 1606.04797. URL: http://arxiv.org/abs/1606.04797.

[72] Ozan Oktay et al. "Attention U-Net: Learning Where to Look for the Pancreas". In: *CoRR* abs/1804.03999 (2018). arXiv: 1804.03999. URL: http://arxiv.org/abs/1804.03999.

[73] Jo Schlemper et al. "Attention gated networks: Learning to leverage salient regions in medical images". In: *Medical Image Analysis* 53 (2019), pp. 197–207. URL: https://www.sciencedirect.com/science/article/pii/S1361841518306133.

[74] Hang Su et al. "Multi-view Convolutional Neural Networks for 3D Shape Recognition". In: *CoRR* abs/1505.00880 (2015). arXiv: 1505.00880. URL: http://arxiv.org/abs/1505.00880.

[75] Daniel Maturana and Sebastian Scherer. "VoxNet: A 3D Convolutional Neural Network for real-time object recognition". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2015, pp. 922–928.

[76] Nima Sedaghat et al. "Orientation-boosted Voxel Nets for 3D Object Recognition". In: *CoRR* abs/1604.03351 (2016). arXiv: 1604.03351. URL: http://arxiv.org/abs/1604.03351.

[77] Shuaifeng Zhi et al. "Toward real-time 3D object recognition: A lightweight volumetric CNN framework using multitask learning". In: *Computers & Graphics* 71 (2018), pp. 199–207. URL: https://www.sciencedirect.com/science/article/pii/S0097849317301735.

[78] Charles Ruizhongtai Qi et al. "Volumetric and Multi-View CNNs for Object Classification on 3D Data". In: *CoRR* abs/1604.03265 (2016). arXiv: 1604.03265. URL: http://arxiv.org/abs/1604.03265.

[79] Vishakh Hegde and Reza Zadeh. "FusionNet: 3D Object Classification Using Multiple Data Representations". In: *CoRR* abs/1607.05695 (2016). arXiv: 1607.05695. URL: http://arxiv.org/abs/1607.05695.

[80] Siarhei Kazlouski. "CT Image Analysis for TB Severity Scoring and CT Report Generation using Autoencoded Image Features". In: *ImageCLEF*. 2019.

[81] Jiancheng Yang et al. "Reinventing 2D Convolutions for 3D Images". In: *IEEE Journal of Biomedical and Health Informatics* 25.8 (Aug. 2021), pp. 3009–3018. URL: http://dx.doi.org/10.1109/JBHI.2021.3049452.

[82] Pedro M Gordaliza et al. "A multi-task self-normalizing 3D-CNN to infer tuberculosis radiological manifestations". In: *arXiv preprint arXiv:1907.12331* (2019).

[83] Sihong Chen et al. "Med3D: Transfer Learning for 3D Medical Image Analysis". In: *CoRR* abs/1904.00625 (2019). arXiv: 1904.00625. URL: http://arxiv.org/abs/1904.00625.

[84] Zhijie Wu et al. "Structure-aware Generative Network for 3D-Shape Modeling". In: *CoRR* abs/1808.03981 (2018). arXiv: 1808.03981. URL: http://arxiv.org/abs/1808.03981.

[85] Andrew Y. Ng and Michael I. Jordan. "On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes". In: *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*. NIPS'01. Vancouver, British Columbia, Canada: MIT Press, 2001, pp. 841–848.

[86] Tom M. Mitchell. *Generative and Discriminative Classifiers: Naive Bayes and Logistic Regression. Machine Learning Classifiers based on Bayes Rule*. McGraw Hill, 2005. URL: www.cs.cmu.edu/%E2%88%BCtom/mlbook.html..

[87] Antonia Creswell et al. "Generative Adversarial Networks: An Overview". In: *CoRR* abs/1710.07035 (2017). arXiv: 1710.07035. URL: http://arxiv.org/abs/1710.07035.

[88] Ian J. Goodfellow. "NIPS 2016 Tutorial: Generative Adversarial Networks". In: *CoRR* abs/1701.00160 (2017). arXiv: 1701.00160. URL: http://arxiv.org/abs/1701.00160.

[89] Padala Manisha and Sujit Gujar. "Generative Adversarial Networks (GANs): What it can generate and What it cannot?" In: *CoRR* abs/1804.00140 (2018). arXiv: 1804.00140. URL: http://arxiv.org/abs/1804.00140.

[90] Ruslan Salakhutdinov et al. "Restricted Boltzmann Machines for Collaborative Filtering". In: *Proceedings of the 24th International Conference on Machine Learning*. ICML '07. Corvalis, Oregon, USA: Association for Computing Machinery, 2007, pp. 791–798. URL: https://doi.org/10.1145/1273496.1273596.

[91] George E. Hinton et al. "A fast learning algorithm for deep belief nets". In: *Neural Computation* 18 (7 2006), pp. 1527–1554. URL: https://pubmed.ncbi.nlm.nih.gov/16764513/.

[92] Diederik P. Kingma and Max Welling. "Auto-Encoding Variational Bayes". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2014. URL: http://arxiv.org/abs/1312.6114.

[93] S. Mohamed and Balaji Lakshminarayanan. "Learning in Implicit Generative Models". In: *ArXiv* abs/1610.03483 (2016).

[94] Alan M. Turing. "Computing Machinery and Intelligence". In: *Mind* 49 (1950), pp. 433–460. URL: https://www.csee.umbc.edu/courses/471/papers/turing.pdf.

[95] Pramod Viswanath. *ECE598: Representation Learning: Algorithms and Models; Lecture 12: Generative Adversarial Network*. Oct. 2016. URL: https://courses.engr.illinois.edu/ece598pv/fa2017/Lecture12_GAN_AmirT.pdf.

[96] Martín Arjovsky et al. "Wasserstein GAN". In: *ArXiv* abs/1701.07875 (2017).

[97] Lilian Weng. "From GAN to WGAN". In: *CoRR* abs/1904.08994 (2019). arXiv: 1904.08994. URL: http://arxiv.org/abs/1904.08994.

[98] William Fedus et al. "Many Paths to Equilibrium: GANs Do Not Need to Decrease a Divergence At Every Step". In: *ArXiv* abs/1710.08446 (2018).

[99] Farzan Farnia and Asuman Ozdaglar. "Do GANs always have Nash equilibria?" In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 13–18 Jul 2020, pp. 3029–3039. URL: http://proceedings.mlr.press/v119/farnia20a.html.

[100] Lars M. Mescheder et al. "The Numerics of GANs". In: *CoRR* abs/1705.10461 (2017). arXiv: 1705.10461. URL: http://arxiv.org/abs/1705.10461.

[101] Vaishnavh Nagarajan and J. Zico Kolter. "Gradient descent GAN optimization is locally stable". In: *CoRR* abs/1706.04156 (2017). arXiv: 1706.04156. URL: http://arxiv.org/abs/1706.04156.

[102] Naveen Kodali et al. "How to Train Your DRAGAN". In: *CoRR* abs/1705.07215 (2017). arXiv: 1705.07215. URL: http://arxiv.org/abs/1705.07215.

[103] Jie Gui et al. "A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications". In: *CoRR* abs/2001.06937 (2020). arXiv: 2001.06937. URL: https://arxiv.org/abs/2001.06937.

[104] Tim Salimans et al. "Improved Techniques for Training GANs". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 2234–2242.

[105] Emily L. Denton et al. "Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks". In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 28. Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper/2015/file/aa169b49b583a2b5af89203c2b78c67c-Paper.pdf.

[106] Xudong Mao et al. "Multi-class Generative Adversarial Networks with the L2 Loss Function". In: *CoRR* abs/1611.04076 (2016). arXiv: 1611.04076. URL: http://arxiv.org/abs/1611.04076.

[107] Ali Borji. "Pros and cons of GAN evaluation measures". In: *Computer Vision and Image Understanding* 179 (2019), pp. 41–65. URL: https://www.sciencedirect.com/science/article/pii/S1077314218304272.

[108] Martin Heusel et al. "GANs Trained by a Two Time-Scale Update Rule Converge to a Nash Equilibrium". In: *CoRR* abs/1706.08500 (2017). arXiv: 1706.08500. URL: http://arxiv.org/abs/1706.08500.

[109] Christian Szegedy et al. “Rethinking the Inception Architecture for Computer Vision”. In: *CoRR* abs/1512.00567 (2015). arXiv: 1512.00567. URL: http://arxiv.org/abs/1512.00567.

[110] Jia Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255.

[111] Yuhuai Wu et al. “On the Quantitative Analysis of Decoder-Based Generative Models”. In: *CoRR* abs/1611.04273 (2016). arXiv: 1611.04273. URL: http://arxiv.org/abs/1611.04273.

[112] Wacha Bounliphone et al. “A test of relative similarity for model selection in generative models”. In: *arXiv preprint arXiv:1511.04581* (2015).

[113] David Lopez-Paz and Maxime Oquab. “Revisiting classifier two-sample tests”. In: *arXiv preprint arXiv:1610.06545* (2016).

[114] Qiantong Xu et al. “An empirical study on evaluation metrics of generative adversarial networks”. In: *CoRR* abs/1806.07755 (2018). arXiv: 1806.07755. URL: http://arxiv.org/abs/1806.07755.

[115] Kaiming He et al. “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition”. In: *CoRR* abs/1406.4729 (2014). arXiv: 1406.4729. URL: http://arxiv.org/abs/1406.4729.

[116] Mehdi Mirza and Simon Osindero. “Conditional Generative Adversarial Nets”. In: *CoRR* abs/1411.1784 (2014). arXiv: 1411.1784. URL: http://arxiv.org/abs/1411.1784.

[117] Justin Johnson et al. “Perceptual Losses for Real-Time Style Transfer and Super-Resolution”. In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe et al. Cham: Springer International Publishing, 2016, pp. 694–711.

[118] Raymond A. Yeh et al. “Semantic Image Inpainting with Perceptual and Contextual Losses”. In: *CoRR* abs/1607.07539 (2016). arXiv: 1607.07539. URL: http://arxiv.org/abs/1607.07539.

[119] Alec Radford et al. “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”. In: *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings* (Nov. 2015). URL: https://arxiv.org/abs/1511.06434v2.

[120] Ankit Yadav and Dinesh Kumar Vishwakarma. “Recent Developments in Generative Adversarial Networks: A Review (Workshop Paper)”. In: *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*. 2020, pp. 404–413.

[121] Vincent Dumoulin et al. “Adversarially learned inference”. In: *arXiv preprint arXiv:1606.00704* (2016).

[122] Jeff Donahue et al. “Adversarial Feature Learning”. In: *CoRR* abs/1605.09782 (2016). arXiv: 1605.09782. URL: http://arxiv.org/abs/1605.09782.

[123] Sebastian Nowozin et al. “f-gan: Training generative neural samplers using variational divergence minimization”. In: *Advances in Neural Information Processing Systems* 29 (2016).

[124] Xi Chen et al. "InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 2180–2188.

[125] Augustus Odena et al. "Conditional image synthesis with auxiliary classifier gans". In: *International conference on machine learning*. PMLR. 2017, pp. 2642–2651.

[126] Han Zhang et al. "StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks". In: *CoRR* abs/1612.03242 (2016). arXiv: 1612.03242. URL: http://arxiv.org/abs/1612.03242.

[127] Xin Yi et al. "Generative Adversarial Network in Medical Imaging: A Review". In: *Medical Image Analysis* 58 (2019), p. 101552.

[128] Phillip Isola et al. "Image-to-Image Translation with Conditional Adversarial Networks". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5967–5976.

[129] Chuan Li and Michael Wand. "Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks". In: *CoRR* abs/1604.04382 (2016). arXiv: 1604.04382. URL: http://arxiv.org/abs/1604.04382.

[130] Jun-Yan Zhu et al. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks". In: *CoRR* abs/1703.10593 (2017). arXiv: 1703.10593. URL: http://arxiv.org/abs/1703.10593.

[131] Yongyi Lu et al. "Conditional CycleGAN for Attribute Guided Face Image Generation". In: *CoRR* abs/1705.09966 (2017). arXiv: 1705.09966. URL: http://arxiv.org/abs/1705.09966.

[132] Ming-Yu Liu et al. "Unsupervised Image-to-Image Translation Networks". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 700–708.

[133] Anders Boesen Lindbo Larsen et al. "Autoencoding beyond pixels using a learned similarity metric". In: *CoRR* abs/1512.09300 (2015). arXiv: 1512.09300. URL: http://arxiv.org/abs/1512.09300.

[134] Cristóbal et al. Esteban. "Real-valued (medical) time series generation with recurrent conditional gans". In: *arXiv preprint arXiv:1706.02633* (2017).

[135] Christian Ledig et al. "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network". In: *CoRR* abs/1609.04802 (2016). arXiv: 1609.04802. URL: http://arxiv.org/abs/1609.04802.

[136] Tero Karras et al. "Progressive Growing of GANs for Improved Quality, Stability, and Variation". In: *CoRR* abs/1710.10196 (2017). arXiv: 1710.10196. URL: http://arxiv.org/abs/1710.10196.

[137] Rameen Abdal et al. "Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space?" In: *CoRR* abs/1904.03189 (2019). arXiv: 1904.03189. URL: http://arxiv.org/abs/1904.03189.

[138] Han Zhang et al. "Self-attention generative adversarial networks". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 7354–7363.

[139] Andrew Brock et al. "Large Scale GAN Training for High Fidelity Natural Image Synthesis". In: *CoRR* abs/1809.11096 (2018). arXiv: 1809.11096. URL: http://arxiv.org/abs/1809.11096.

[140] Taeksoo Kim et al. "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks". In: *CoRR* abs/1703.05192 (2017). arXiv: 1703.05192. URL: http://arxiv.org/abs/1703.05192.

[141] Karim Armanious et al. "MedGAN: Medical Image Translation using GANs". In: *CoRR* abs/1806.06397 (2018). arXiv: 1806.06397. URL: http://arxiv.org/abs/1806.06397.

[142] Xin Yu and Fatih Porikli. "Ultra-Resolving Face Images by Discriminative Generative Networks". In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe et al. Cham: Springer International Publishing, 2016, pp. 318–333.

[143] Raeid Saqur and Sal Vivona. "CapsGAN: Using Dynamic Routing for Generative Adversarial Networks". In: *CoRR* abs/1806.03968 (2018). arXiv: 1806.03968. URL: http://arxiv.org/abs/1806.03968.

[144] Andrew Brock et al. "Neural Photo Editing with Introspective Adversarial Networks". In: *CoRR* abs/1609.07093 (2016). arXiv: 1609.07093. URL: http://arxiv.org/abs/1609.07093.

[145] Scott E. Reed et al. "Generative Adversarial Text to Image Synthesis". In: *CoRR* abs/1605.05396 (2016). arXiv: 1605.05396. URL: http://arxiv.org/abs/1605.05396.

[146] Gerhard Paul Diller et al. "Utility of deep learning networks for the generation of artificial cardiac magnetic resonance images in congenital heart disease". In: *BMC Medical Imaging* 20 (1 Oct. 2020), p. 113. URL: https://bmcmedimaging.biomedcentral.com/articles/10.1186/s12880-020-00511-1.

[147] Pedro Costa et al. "End-to-End Adversarial Retinal Image Synthesis". In: *IEEE Transactions on Medical Imaging* 37.3 (2018), pp. 781–791.

[148] Maayan Frid-Adar et al. "GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification". In: *CoRR* abs/1803.01229 (2018). arXiv: 1803.01229. URL: http://arxiv.org/abs/1803.01229.

[149] Semen E. Konstein and Igor V. Tyutin. "Ideals generated by traces or by supertraces in the symplectic reflection algebra H1,ν (I2(2m+1))". In: *Journal of Nonlinear Mathematical Physics* 24 (2016), pp. 405–425.

[150] Jelmer M. Wolterink et al. "Deep MR to CT Synthesis using Unpaired Data". In: *CoRR* abs/1708.01155 (2017). arXiv: 1708.01155. URL: http://arxiv.org/abs/1708.01155.

[151] Zizhao Zhang et al. "Translating and Segmenting Multimodal Medical Volumes with Cycle- and Shape-Consistency Generative Adversarial Network". In: *CoRR* abs/1802.09655 (2018). arXiv: 1802.09655. URL: http://arxiv.org/abs/1802.09655.

[152] Agisilaos Chartsias et al. "Adversarial Image Synthesis for Unpaired Multi-modal Cardiac Data". In: *Simulation and Synthesis in Medical Imaging*. Ed. by Sotirios A. Tsaftaris et al. Cham: Springer International Publishing, 2017, pp. 3–13.

[153] Joseph Paul Cohen et al. "Distribution Matching Losses Can Hallucinate Features in Medical Image Translation". In: *CoRR* abs/1805.08841 (2018). arXiv: 1805.08841. URL: http://arxiv.org/abs/1805.08841.

[154] Agisilaos Chartsias et al. "Factorised Representation Learning in Cardiac Image Analysis". In: *CoRR* abs/1903.09467 (2019). arXiv: 1903.09467. URL: http://arxiv.org/abs/1903.09467.

[155] Felix Lau et al. "ScarGAN: Chained Generative Adversarial Networks to Simulate Pathological Tissue on Cardiovascular MR Scans". In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Ed. by Danail Stoyanov et al. Cham: Springer International Publishing, 2018, pp. 343–350.

[156] Yan Xia et al. "Recovering from missing data in population imaging – Cardiac MR image imputation via conditional generative adversarial nets". In: *Medical Image Analysis* 67 (2021), p. 101812. URL: https://www.sciencedirect.com/science/article/pii/S1361841520301766.

[157] Xin Yi and Paul S. Babyn. "Sharpness-aware Low dose CT denoising using conditional generative adversarial network". In: *CoRR* abs/1708.06453 (2017). arXiv: 1708.06453. URL: http://arxiv.org/abs/1708.06453.

[158] Ki Hwan Kim et al. "Improving resolution of MR images with an adversarial network incorporating images with different contrast". In: *Medical Physics* 45 (7 July 2018), pp. 3120–3131. URL: https://pubmed.ncbi.nlm.nih.gov/29729006/.

[159] Yan Wang et al. "3D conditional generative adversarial networks for high-quality PET image estimation at low dose". In: *NeuroImage* 174 (July 2018), pp. 550–562. URL: https://pubmed.ncbi.nlm.nih.gov/29571715/.

[160] Guang Yang et al. "DAGAN: Deep De-Aliasing Generative Adversarial Networks for Fast Compressed Sensing MRI Reconstruction". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1310–1321.

[161] Ilkay Oksuz et al. "Cardiac MR Motion Artefact Correction from K-space Using Deep Learning-Based Reconstruction". In: *Machine Learning for Medical Image Reconstruction*. Ed. by Florian Knoll et al. Cham: Springer International Publishing, 2018, pp. 21–29.

[162] Mina Rezaei et al. "Whole Heart and Great Vessel Segmentation with Context-aware of Generative Adversarial Networks". In: *Bildverarbeitung für die Medizin 2018*. Ed. by Andreas Maier et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, pp. 353–358.

[163] Yuan Xue et al. "SegAN: Adversarial Network with Multi-scale L_1 Loss for Medical Image Segmentation". In: *CoRR* abs/1706.01805 (2017). arXiv: 1706.01805. URL: http://arxiv.org/abs/1706.01805.

[164] Mina Rezaei et al. "Conditional Generative Refinement Adversarial Networks for Unbalanced Medical Image Semantic Segmentation". In: *CoRR* abs/1810.03871 (2018). arXiv: 1810.03871. URL: http://arxiv.org/abs/1810.03871.

[165] Wentao Zhu et al. "Adversarial Deep Structured Nets for Mass Segmentation from Mammograms". In: *CoRR* abs/1710.09288 (2017). arXiv: 1710.09288. URL: http://arxiv.org/abs/1710.09288.

[166] Wei Dai et al. "SCAN: Structure Correcting Adversarial Network for Chest X-rays Organ Segmentation". In: *CoRR* abs/1703.08770 (2017). arXiv: 1703.08770. URL: http://arxiv.org/abs/1703.08770.

[167] Konstantinos Kamnitsas et al. "Unsupervised domain adaptation in brain lesion segmentation with adversarial networks". In: *CoRR* abs/1612.08894 (2016). arXiv: 1612.08894. URL: http://arxiv.org/abs/1612.08894.

[168] Xin Yi et al. "Unsupervised and semi-supervised learning with Categorical Generative Adversarial Networks assisted by Wasserstein distance for dermoscopy image Classification". In: *CoRR* abs/1804.03700 (2018). arXiv: 1804.03700. URL: http://arxiv.org/abs/1804.03700.

[169] Alireza Makhzani et al. "Adversarial Autoencoders". In: *CoRR* abs/1511.05644 (2015). arXiv: 1511.05644. URL: http://arxiv.org/abs/1511.05644.

[170] Carl Doersch. "Tutorial on variational autoencoders". In: *arXiv preprint arXiv:1606.05908* (2016).

[171] Jeremy Jordan. *Variational Autoencoders.* URL: https://www.jeremyjordan.me/variational-autoencoders/ (visited on 08/08/2021).

[172] Ruoqi Wei et al. "Variations in Variational Autoencoders - A Comparative Evaluation". In: *IEEE Access* 8 (2020), pp. 153651–153670.

[173] Lilian Weng. *From Autoencoder to Beta-VAE.* URL: https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html#vae-variational-autoencoder (visited on 08/08/2021).

[174] Joseph Rocca. *Understanding Variational Autoencoders (VAEs).* URL: https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73 (visited on 08/08/2021).

[175] Ishaan Gulrajani et al. "PixelVAE: A Latent Variable Model for Natural Images". In: *CoRR* abs/1611.05013 (2016). arXiv: 1611.05013. URL: http://arxiv.org/abs/1611.05013.

[176] Durk P. Kingma et al. "Variational Dropout and the Local Reparameterization Trick". In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 28. Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper/2015/file/bc7316929fe1545bf0b98d114ee3ecb8-Paper.pdf.

[177] Lucas Theis et al. "A note on the evaluation of generative models". In: *arXiv preprint arXiv:1511.01844* (2015).

[178] Ferenc Huszár. "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" In: *arXiv preprint arXiv:1511.05101* (2015).

[179] Serena Yeung et al. "Tackling Over-pruning in Variational Autoencoders". In: *CoRR* abs/1706.03643 (2017). arXiv: 1706.03643. URL: http://arxiv.org/abs/1706.03643.

[180] Kihyuk Sohn et al. "Learning Structured Output Representation using Deep Conditional Generative Models". In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 28. Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper/2015/file/8d55a249e6baa5c06772297520da2051-Paper.pdf.

[181] Aäron van den Oord et al. "Neural Discrete Representation Learning". In: *CoRR* abs/1711.00937 (2017). arXiv: 1711.00937. URL: http://arxiv.org/abs/1711.00937.

[182] Shengjia Zhao et al. "InfoVAE: Information Maximizing Variational Autoencoders". In: *CoRR* abs/1706.02262 (2017). arXiv: 1706.02262. URL: http://arxiv.org/abs/1706.02262.

[183] Irina Higgins et al. "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". In: *ICLR*. 2017.

[184] N. Siddharth et al. "Learning Disentangled Representations with Semi-Supervised Deep Generative Models". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 5927–5937.

[185] Tim R. Davidson et al. "Hyperspherical variational auto-encoders". In: *arXiv preprint arXiv:1804.00891* (2018).

[186] Carlo Biffi et al. "Explainable anatomical shape analysis through deep hierarchical generative models". In: *IEEE Transactions on Medical Imaging* 39.6 (2020), pp. 2088–2099.

[187] Casper Kaae Sønderby et al. "Ladder Variational Autoencoders". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 3745–3753.

[188] Caroline Petitjean and Jean-Nicolas Dacher. "A review of segmentation methods in short axis cardiac MR images". In: *Medical Image Analysis* 15.2 (2011), pp. 169–184. URL: https://www.sciencedirect.com/science/article/pii/S1361841510001349.

[189] Chris A. Cocosco et al. "Automatic cardiac region-of-interest computation in cine 3D structural MRI". In: *International Congress Series* 1268 (2004). CARS 2004 - Computer Assisted Radiology and Surgery. Proceedings of the 18th International Congress and Exhibition, pp. 1126–1131. URL: https://www.sciencedirect.com/science/article/pii/S0531513104006089.

[190] Amin Katouzian et al. "A new automated technique for left-and right-ventricular segmentation in magnetic resonance imaging". In: *Conference proceedings : Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference* 2006 (2006), pp. 3074–3077. URL: https://pubmed.ncbi.nlm.nih.gov/17946545/.

[191] YingLi Lu et al. "Segmentation of Left Ventricle in Cardiac Cine MRI: An Automatic Image-Driven Method". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5528 (2009), pp. 339–347. URL: https://link.springer.com/chapter/10.1007/978-3-642-01932-6_37.

[192] J. Y. Yeh et al. "Myocardial border detection by branch-and-bound dynamic programming in magnetic resonance images". In: *Computer Methods and Programs in Biomedicine* 79 (1 July 2005), pp. 19–29.

[193] Jean Cousty et al. "Segmentation of 4D cardiac MRI: Automated method based on spatio-temporal watershed cuts". In: *Image and Vision Computing* 28 (8 Aug. 2010), pp. 1229–1243.

[194] Michael Kass et al. "Snakes: Active contour models". In: *International Journal of Computer Vision 1987 1:4* 1 (4 Jan. 1988), pp. 321–331. URL: https://link.springer.com/article/10.1007/BF00133570.

[195] S. Ranganath. "Contour extraction from cardiac MRI studies using snakes". In: *IEEE Transactions on Medical Imaging* 14 (2 1995), pp. 328–338. URL: https://pubmed.ncbi.nlm.nih.gov/18215836/.

[196] T.F. Cootes et al. "Active Shape Models-Their Training and Application". In: *Computer Vision and Image Understanding* 61.1 (1995), pp. 38–59. URL: https://www.sciencedirect.com/science/article/pii/S1077314285710041.

[197] Steven C. Mitchell et al. "Segmentation of cardiac MR images: an active appearance model approach". In: *Medical Imaging 2000: Image Processing*. Ed. by Kenneth M. Hanson. Vol. 3979. International Society for Optics and Photonics. SPIE, 2000, pp. 224–234. URL: https://doi.org/10.1117/12.387684.

[198] Torsten Rohlfing et al. "Quo Vadis, Atlas-Based Segmentation?" In: *Handbook of Biomedical Image Analysis: Volume III: Registration Models*. Ed. by Jasjit S. Suri et al. Boston, MA: Springer US, 2005, pp. 435–486. URL: https://doi.org/10.1007/0-306-48608-3_11.

[199] Maria Lorenzo-Valdés et al. "Segmentation of 4D cardiac MR images using a probabilistic atlas and the EM algorithm". In: *Medical Image Analysis* 8.3 (2004), pp. 255–265. URL: https://www.sciencedirect.com/science/article/pii/S1361841504000271.

[200] Rahil Shahzad et al. "Fully-automatic left ventricular segmentation from long-axis cardiac cine MR scans". In: *Medical Image Analysis* 39 (2017), pp. 44–55. URL: https://www.sciencedirect.com/science/article/pii/S1361841517300567.

[201] Olivier Bernard et al. "Deep Learning Techniques for Automatic MRI Cardiac Multi-Structures Segmentation and Diagnosis: Is the Problem Solved?" In: *IEEE Transactions on Medical Imaging* 37.11 (2018), pp. 2514–2525.

[202] M.R. Avendi et al. "A combined deep-learning and deformable-model approach to fully automatic segmentation of the left ventricle in cardiac MRI". In: *Medical Image Analysis* 30 (2016), pp. 108–119. URL: https://www.sciencedirect.com/science/article/pii/S1361841516000128.

[203] Phi Vu Tran. "A Fully Convolutional Neural Network for Cardiac Segmentation in Short-Axis MRI". In: *CoRR* abs/1604.00494 (2016). arXiv: 1604.00494. URL: http://arxiv.org/abs/1604.00494.

[204] Wenjia Bai et al. "Automated cardiovascular magnetic resonance image analysis with fully convolutional networks". In: *Journal of Cardiovascular Magnetic Resonance 2018 20:1* 20 (1 Sept. 2018), pp. 1–12. URL: https://jcmr-online.biomedcentral.com/articles/10.1186/s12968-018-0471-x.

[205] Ozan Oktay et al. "Anatomically Constrained Neural Networks (ACNN): Application to Cardiac Image Enhancement and Segmentation". In: *CoRR* abs/1705.08302 (2017). arXiv: 1705.08302. URL: http://arxiv.org/abs/1705.08302.

[206] Jun Zhang et al. "LU-NET: An Improved U-Net for Ventricular Segmentation". In: *IEEE Access* 7 (2019), pp. 92539–92546.

[207] Jinming Duan et al. "Automatic 3D bi-ventricular segmentation of cardiac images by a shape-constrained multi-task deep learning approach". In: *CoRR* abs/1808.08578 (2018). arXiv: 1808.08578. URL: http://arxiv.org/abs/1808.08578.

[208] Zhanwei Xu et al. "CFUN: Combining Faster R-CNN and U-net Network for Efficient Whole Heart Segmentation". In: *CoRR* abs/1812.04914 (2018). arXiv: 1812.04914. URL: http://arxiv.org/abs/1812.04914.

[209] Chengqin Ye et al. "Multi-Depth Fusion Network for Whole-Heart CT Image Segmentation". In: *IEEE Access* 7 (2019), pp. 23421–23429.

[210] Chenchu Xu et al. "MuTGAN: Simultaneous Segmentation and Quantification of Myocardial Infarction Without Contrast Agents via Joint Adversarial Learning". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*. Ed. by Alejandro F. Frangi et al. Cham: Springer International Publishing, 2018, pp. 525–534.

[211] Wufeng Xue et al. "Full left ventricle quantification via deep multitask relationships learning". In: *Medical Image Analysis* 43 (2018), pp. 54–65. URL: https://www.sciencedirect.com/science/article/pii/S1361841517301366.

[212] Alejandro Debus and Enzo Ferrante. "Left ventricle quantification through spatio-temporal CNNs". In: *CoRR* abs/1808.07967 (2018). arXiv: 1808.07967. URL: http://arxiv.org/abs/1808.07967.

[213] Bin Kong et al. "Recognizing End-Diastole and End-Systole Frames via Deep Temporal Regression Network". In: *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016*. Ed. by Sebastien Ourselin et al. Cham: Springer International Publishing, 2016, pp. 264–272.

[214] Mariam Afshin et al. "Regional Assessment of Cardiac Left Ventricular Myocardial Function via MRI Statistical Features". In: *IEEE Transactions on Medical Imaging* 33.2 (2014), pp. 481–494.

[215] Mantilla J. et al. "Classification of LV wall motion in cardiac MRI using kernel Dictionary Learning with a parametric approach". In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference* 2015 (Nov. 2015), pp. 7292–7295. URL: https://pubmed.ncbi.nlm.nih.gov/26737975/.

[216] Z. Qian et al. "Identifying regional cardiac abnormalities from myocardial strains using spatio-temporal tensor analysis". In: *Medical Image Computing and Computer-Assisted Intervention : MICCAI* 11 (Pt 1 2008), pp. 789–797. URL: https://pubmed.ncbi.nlm.nih.gov/18979818/.

[217] Nicolas Duchateau et al. "A spatiotemporal statistical atlas of motion for the quantification of abnormal myocardial tissue velocities". In: *Medical Image Analysis* 15 (3 June 2011), pp. 316–328. URL: https://pubmed.ncbi.nlm.nih.gov/21315650/.

[218] Wenjia Bai et al. "A bi-ventricular cardiac atlas built from 1000+ high resolution MR images of healthy subjects and an analysis of shape and motion". In: *Medical Image Analysis* 26 (1 2015), pp. 133–145. URL: https://pubmed.ncbi.nlm.nih.gov/26387054/.

[219] Wufeng Xue et al. "Cardiac Motion Scoring with Segment- and Subject-level Non-Local Modeling". In: *CoRR* abs/1806.05569 (2018). arXiv: 1806.05569. URL: http://arxiv.org/abs/1806.05569.

[220] Xiaolong Wang et al. "Non-local Neural Networks". In: *CoRR* abs/1711.07971 (2017). arXiv: 1711.07971. URL: http://arxiv.org/abs/1711.07971.

[221] Chen Qin et al. "Biomechanics-informed neural networks for myocardial motion tracking in MRI". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2020, pp. 296–306.

[222] Yunxuan Zhang et al. "CMR Motion Artifact Correction using Generative Adversarial Nets". In: *CoRR* abs/1902.11121 (2019). arXiv: 1902.11121. URL: http://arxiv.org/abs/1902.11121.

[223] Chen Qin et al. "Joint Learning of Motion Estimation and Segmentation for Cardiac MR Image Sequences". In: *CoRR* abs/1806.04066 (2018). arXiv: 1806.04066. URL: http://arxiv.org/abs/1806.04066.

[224] Wenjun Yan et al. "Cine MRI analysis by deep learning of optical flow: Adding the temporal dimension". In: *Computers in Biology and Medicine* 111 (2019), p. 103356. URL: https://www.sciencedirect.com/science/article/pii/S0010482519302331.

[225] Julian Krebs et al. "Probabilistic Motion Modeling from Medical Image Sequences: Application to Cardiac Cine-MRI". In: *CoRR* abs/1907.13524 (2019). arXiv: 1907.13524. URL: http://arxiv.org/abs/1907.13524.

[226] Julian Krebs et al. "Learning Structured Deformations using Diffeomorphic Registration". In: *CoRR* abs/1804.07172 (2018). arXiv: 1804.07172. URL: http://arxiv.org/abs/1804.07172.

[227] Ping Lu et al. "Going Deeper into Cardiac Motion Analysis to Model Fine Spatio-Temporal Features". In: *Medical Image Understanding and Analysis*. Ed. by Bartłomiej W. Papież et al. Cham: Springer International Publishing, 2020, pp. 294–306.

[228] Rahman Attar et al. "3D cardiac shape prediction with deep neural networks: simultaneous use of images and patient metadata". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2019, pp. 586–594.

[229] Ethan Perez et al. "FiLM: Visual Reasoning with a General Conditioning Layer". In: *CoRR* abs/1709.07871 (2017). arXiv: 1709.07871. URL: http://arxiv.org/abs/1709.07871.

[230] Ghalib Bello et al. "Deep-learning cardiac motion analysis for human survival prediction". In: *Nature Machine Intelligence* 1 (Feb. 2019), pp. 95–104.

[231] Ali Madani et al. "Deep echocardiography: data-efficient supervised and semi-supervised deep learning towards automated diagnosis of cardiac disease". In: *NPJ Digital Medicine 2018* 1 (1 Oct. 2018), pp. 1–11. URL: https://www.nature.com/articles/s41746-018-0065-x.

[232] Mingqiang Chen et al. "Deep Learning Assessment of Myocardial Infarction From MR Image Sequences". In: *IEEE Access* 7 (2019), pp. 5438–5446.

[233] Chenchu Xu et al. "Direct detection of pixel-level myocardial infarction areas via a deep-learning algorithm". In: *CoRR* abs/1706.03182 (2017). arXiv: 1706.03182. URL: http://arxiv.org/abs/1706.03182.

[234] Fabian Isensee et al. "Automatic Cardiac Disease Assessment on cine-MRI via Time-Series Segmentation and Domain Specific Features". In: *CoRR* abs/1707.00587 (2017). arXiv: 1707.00587. URL: http://arxiv.org/abs/1707.00587.

[235] Dariusz Góra. "The Pierre Auger Observatory: Review of Latest Results and Perspectives". In: *Universe* 4.11 (Nov. 2018), p. 128. URL: http://dx.doi.org/10.3390/universe4110128.

[236] Carlo Biffi et al. "Learning Interpretable Anatomical Features Through Deep Generative Models: Application to Cardiac Remodeling". In: *CoRR* abs/1807.06843 (2018). arXiv: 1807.06843. URL: http://arxiv.org/abs/1807.06843.

[237] Maxime Sermesant et al. "Applications of artificial intelligence in cardiovascular imaging". In: *Nature Reviews. Cardiology* 18 (8 Aug. 2021), pp. 600–609. URL: https://pubmed.ncbi.nlm.nih.gov/33712806/.

[238] Gaëtan Desrues et al. "Personal-by-design: a 3D Electromechanical Model of the Heart Tailored for Personalisation". In: *Functional Imaging and Modeling of the Heart* (2021). URL: https://hal.inria.fr/hal-03369345.

[239] Kaggle. *Second Annual Data Science Bowl*. URL: https://www.kaggle.com/c/second-annual-data-science-bowl/overview (visited on 08/30/2021).

[240] Yann A. LeCun et al. "A tutorial on energy-based learning". English (US). In: *Predicting Structured Data*. Ed. by G. Bakir et al. MIT Press, 2006.

[241] Junbo Jake Zhao et al. "Energy-based Generative Adversarial Network". In: *CoRR* abs/1609.03126 (2016). arXiv: 1609.03126. URL: http://arxiv.org/abs/1609.03126.

[242] Qi Mao et al. "Mode Seeking Generative Adversarial Networks for Diverse Image Synthesis". In: *CoRR* abs/1903.05628 (2019). arXiv: 1903.05628. URL: http://arxiv.org/abs/1903.05628.

[243] Sen Pei et al. "Alleviating Mode Collapse in GAN via Diversity Penalty Module". In: *CoRR* abs/2108.02353 (2021). arXiv: 2108.02353. URL: https://arxiv.org/abs/2108.02353.

[244] Julius Ossenberg-Engels and Vicente Grau. "Conditional Generative Adversarial Networks for the Prediction of Cardiac Contraction from Individual Frames". In: *International Workshop on Statistical Atlases and Computational Models of the Heart*. 2019, pp. 109–118.

[245] Vicente Bodi et al. "Prognostic value of a comprehensive cardiac magnetic resonance assessment soon after a first ST-segment elevation myocardial infarction". In: *JACC. Cardiovascular imaging* 2 (7 July 2009), pp. 835–842. URL: https://pubmed.ncbi.nlm.nih.gov/19608133/.

[246] Cathie Sudlow et al. "UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age". In: *PLOS Medicine* 12.3 (2015), pp. 1–10. URL: https://doi.org/10.1371/journal.pmed.1001779.

[247] Constantinos O'Mahony et al. "A novel clinical risk prediction model for sudden cardiac death in hypertrophic cardiomyopathy (HCM risk-SCD)". In: *European Heart Journal* 35 (30 Aug. 2014), pp. 2010–2020. URL: https://pubmed.ncbi.nlm.nih.gov/24126876/.

[248] Marcel J.M. Kofflard et al. "Hypertrophic cardiomyopathy in a large community-based population: Clinical outcome and identification of risk factors for sudden cardiac death and clinical deterioration". In: *Journal of the American College of Cardiology* 41 (6 Mar. 2003), pp. 987–993.

[249] Sanae Maki et al. "Predictors of sudden cardiac death in hypertrophic cardiomyopathy". In: *American Journal of Cardiology* 82 (6 A 1998), pp. 774–778.

[250] Julia V. Montgomery et al. "Relation of electrocardiographic patterns to phenotypic expression and clinical outcome in hypertrophic cardiomyopathy". In: *American Journal of Cardiology* 96 (2 July 2005), pp. 270–275.

[251] Taku Inohara et al. "Association of of atrial fibrillation clinical phenotypes with treatment patterns and outcomes a multicenter registry study". In: *JAMA Cardiology* 3 (1 Jan. 2018), pp. 54–63.

[252] Aurore Lyon et al. "Distinct ECG phenotypes identified in hypertrophic cardiomyopathy using machine learning associate with arrhythmic risk markers". In: *Frontiers in Physiology* 9 (MAR Mar. 2018).

[253] Dai Yin Lu et al. "E / e' ratio and outcome prediction in hypertrophic cardiomyopathy: The influence of outflow tract obstruction". In: *European Heart Journal Cardiovascular Imaging* 19 (1 Jan. 2018), pp. 101–107.

[254] Quirino Ciampi et al. "Prognostic role of stress echocardiography in hypertrophic cardiomyopathy: The International Stress Echo Registry". In: *International Journal of Cardiology* 219 (Sept. 2016), pp. 331–338.

[255] Betty Raman et al. "Progression of myocardial fibrosis in hypertrophic cardiomyopathy: Mechanisms and clinical implications". In: *European Heart Journal Cardiovascular Imaging* 20 (2 Feb. 2019), pp. 157–167.

[256] American Heart Association Editorial Staff. *Ejection Fraction Heart Failure Measurement*. URL: https://www.heart.org/en/health-topics/heart-failure/diagnosing-heart-failure/ejection-fraction-heart-failure-measurement (visited on 08/21/2021).

[257] Steffen E. Petersen et al. "Reference ranges for cardiac structure and function using cardiovascular magnetic resonance (CMR) in Caucasians from the UK Biobank population cohort". In: *Journal of Cardiovascular Magnetic Resonance* 19.1 (2017), p. 18.

[258] Armin Arbab-Zadeh et al. "Effect of Aging and Physical Activity on Left Ventricular Compliance". In: *Circulation* 110 (13 Sept. 2004), pp. 1799–1805. URL: https://www.ahajournals.org/doi/abs/10.1161/01.cir.0000142863.71285.74.

[259] James B. Strait and Edward G. Lakatta. "Aging-associated cardiovascular changes and their relationship to heart failure". In: *Heart Failure Clinics* 8 (1 Jan. 2012), p. 143. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3223374/.

[260] Mahmoud Mostapha et al. "Braided Networks for Scan-Aware MRI Brain Tissue Segmentation". In: *Proceedings - International Symposium on Biomedical Imaging* 2020-April (Apr. 2020), pp. 136–139.

[261] Jinsung Yoon et al. "GAIN: Missing Data Imputation using Generative Adversarial Nets". In: *CoRR* abs/1806.02920 (2018). arXiv: 1806.02920. URL: http://arxiv.org/abs/1806.02920.

[262] Chen Chen et al. "Deep Learning for Cardiac Image Segmentation: A Review". In: *Frontiers in Cardiovascular Medicine* 7 (Mar. 2020).

[263] Christian F. Baumgartner et al. "An Exploration of 2D and 3D Deep Learning Techniques for Cardiac MR Image Segmentation". In: *CoRR* abs/1709.04496 (2017). arXiv: 1709.04496. URL: http://arxiv.org/abs/1709.04496.

[264] Sulaiman Vesal et al. "Fully Automated 3D Cardiac MRI Localisation and Segmentation Using Deep Neural Networks". In: *Journal of Imaging* 6 (July 2020), p. 65.

[265] Qihang Yu et al. "Thickened 2D Networks for 3D Medical Image Segmentation". In: *CoRR* abs/1904.01150 (2019). arXiv: 1904.01150. URL: http://arxiv.org/abs/1904.01150.

[266] Abhirup Banerjee et al. "A completely automated pipeline for 3D reconstruction of human heart from 2D cine magnetic resonance slices". In: *Philosophical Transactions of the Royal Society A* 379 (2212 Dec. 2021). URL: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2020.0257.

[267] Maayan Frid-Adar et al. "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification". In: *Neurocomputing* 321 (2018), pp. 321–331. URL: https://www.sciencedirect.com/science/article/pii/S0925231218310749.

[268] Youssef Skandarani et al. "Gans for medical image synthesis: An empirical study". In: *arXiv preprint arXiv:2105.05318* (2021).

[269] Hao Xu et al. "Calculation of Anatomical and Functional Metrics Using Deep Learning in Cardiac MRI: Comparison Between Direct and Segmentation-Based Estimation". In: *Statistical Atlases and Computational Models of the Heart. Atrial Segmentation and LV Quantification Challenges*. Ed. by Mihaela Pop et al. Cham: Springer International Publishing, 2019, pp. 402–411.

[270] Lei Li et al. "Joint Left Atrial Segmentation and Scar Quantification Based on a DNN with Spatial Encoding and Shape Attention". In: *Medical Image Computing and Computer Assisted Intervention*. Lima, Peru: Springer-Verlag, 2020, pp. 118–127. URL: https://doi.org/10.1007/978-3-030-59719-1_12.

[271] Daniela O. Medley et al. "Deep Active Shape Model for Robust Object Fitting". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 2380–2394.

[272] Ishan Poudel et al. "Major Adverse Cardiovascular Events: An Inevitable Outcome of ST-elevation myocardial infarction? A Literature Review". In: *Cureus* 11 (7 July 2019). URL: https://www.cureus.com/articles/21561-major-adverse-cardiovascular-events-an-inevitable-outcome-of-st-elevation-myocardial-infarction-a-literature-review.

[273] Borja Ibanez et al. "2017 ESC Guidelines for the management of acute myocardial infarction in patients presenting with ST-segment elevation". In: *European Heart Journal* 39 (2 Jan. 2018), pp. 119–177.

[274] Jean Philippe Collet et al. "2020 ESC Guidelines for the management of acute coronary syndromes in patients presenting without persistent ST-segment elevation". In: *European Heart Journal* 42 (14 Apr. 2021), pp. 1289–1367.

[275] Holger Thiele et al. "Effect of aspiration thrombectomy on microvascular obstruction in NSTEMI patients: the TATORT-NSTEMI trial". In: *Journal of the American College of Cardiology* 64 (11 2014), pp. 1117–1124. URL: https://pubmed.ncbi.nlm.nih.gov/25212646/.

[276] Ingo Eitel et al. "Intracoronary compared with intravenous bolus abciximab application during primary percutaneous coronary intervention in ST-segment elevation myocardial infarction: cardiac magnetic resonance substudy of the AIDA STEMI trial". In: *Journal of the American College of Cardiology* 61 (13 Apr. 2013), pp. 1447–1454. URL: https://pubmed.ncbi.nlm.nih.gov/23466078/.

[277] Jorge Corral Acero et al. "SMOD - Data Augmentation Based on Statistical Models of Deformation to Enhance Segmentation in 2D Cine Cardiac MRI". In: *Functional Imaging and Modeling of the Heart - 10th International Conference, FIMH 2019, Proceedings*. Ed. by Yves Coudière et al. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Germany: Springer Verlag, May 2019, pp. 361–369.

[278] Jorge Corral Acero et al. "Left Ventricle Quantification with Cardiac MRI: Deep Learning Meets Statistical Models of Deformation". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12009 LNCS (2020), pp. 384–394.

[279] Pablo Lamata et al. "An automatic service for the personalization of ventricular cardiac meshes". In: *Journal of the Royal Society Interface* 11 (91 Feb. 2014). URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3869175/.

[280] Jorge Corral Acero et al. "Understanding and Improving Risk Assessment after Infarction using Automated Left Ventricular Shape Analysis". In: *JACC Cardiovascular Imaging (Under 3rd Revision)* (Under Review).

[281] Andreas Schuster et al. "Fast manual long-axis strain assessment provides optimized cardiovascular event prediction following myocardial infarction". In: *European Heart Journal - Cardiovascular Imaging* 20 (11 Nov. 2019), pp. 1262–1270. URL: https://academic.oup.com/ehjcimaging/article/20/11/1262/5481121.

[282] Agnes Mayr et al. "Mitral annular plane systolic excursion by cardiac MR is an easy tool for optimized prognosis assessment in ST-elevation myocardial infarction". In: *European Radiology* 30 (1 Jan. 2020), p. 620. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6890588/.

[283] José T. Ortiz-Pérez et al. "Correspondence between the 17-segment model and coronary arterial anatomy using contrast-enhanced cardiac magnetic resonance imaging". In: *JACC. Cardiovascular imaging* 1 (3 2008), pp. 282–293. URL: https://pubmed.ncbi.nlm.nih.gov/19356440/.

[284] Adrianna Janik et al. "Interpretability of a Deep Learning Model in the Application of Cardiac MRI Segmentation with an ACDC Challenge Dataset". In: *CoRR* abs/2103.08590 (2021). arXiv: 2103.08590. URL: https://arxiv.org/abs/2103.08590.

[285] Been Kim et al. "Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV)". In: *35th International Conference on Machine Learning, ICML 2018* 6 (Nov. 2017), pp. 4186–4195. URL: https://arxiv.org/abs/1711.11279v5.

[286] Aditya Chattopadhyay et al. "Grad-CAM++: Generalized Gradient-based Visual Explanations for Deep Convolutional Networks". In: *CoRR* abs/1710.11063 (2017). arXiv: 1710.11063. URL: http://arxiv.org/abs/1710.11063.

[287] Thomas Stiermaier et al. "Cardiac Magnetic Resonance Left Ventricular Mechanical Uniformity Alterations for Risk Assessment After Acute Myocardial Infarction". In: *Journal of the American Heart Association* 8 (16 Aug. 2019). URL: https://www.ahajournals.org/doi/abs/10.1161/JAHA.118.011576.

[288] Franco Scarselli et al. "The graph neural network model". In: *IEEE Transactions on Neural Networks* 20 (1 Jan. 2009), pp. 61–80. URL: https://pubmed.ncbi.nlm.nih.gov/19068426/.

[289] Marcel Beetz et al. "Biventricular Surface Reconstruction From Cine Mri Contours Using Point Completion Networks". In: *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. 2021, pp. 105–109.

[290] Wentao Yuan et al. “PCN: Point Completion Network”. In: *CoRR* abs/1808.00671 (2018). arXiv: 1808.00671. URL: http://arxiv.org/abs/1808.00671.

[291] Charles Ruizhongtai Qi et al. “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space”. In: *CoRR* abs/1706.02413 (2017). arXiv: 1706.02413. URL: http://arxiv.org/abs/1706.02413.