Imperial College London

BioMedIA

Shape Analysis using Graph Neural Networks in Medical Imaging

Author Nairouz Shehata Hamed Mohamed [1,4]

Submitted April 2025

Supervisors Prof. Ben Glocker[1]
Dr. Wenjia Bai[1]
Dr.-Ing Heba Aguib[2]
Prof. Sir Magdi Yacoub[3,4]

[1]Department of Computing, Imperial College London, UK
[2]BMW Foundation Herbet Quandt, Munich, Germany
[3]National Health & Lung Institute, Imperial College London, UK
[4]Aswan Heart Centre, Aswan, Egypt

A thesis submitted in fulfilment of the requirements for the degree of
Philosophiae Doctor

Biomedical Image Analysis Group, Department of Computing,
Imperial College London

Biomedical Image Analysis (BioMedIA) Group,
Department of Computing,
Huxley Building,
Imperial College London,
180 Queens Gate,
South Kensington,
SW7 2AZ

Nairouz Shehata © 2025

”A dream you dream alone is only a dream. A dream you dream together is reality” .

John Lennon - *singer, songwriter & member of The Beatles*

.The algorithm is YOU.

To my family, friends, peers, and the inspiring mentors and leaders who guided me along the way.

Declaration

I, Nairouz Shehata, hereby declare that this thesis is my own work unless otherwise specified. All published papers that are reproduced in this thesis are appropriately referenced and the reproduction adheres to the relevant copyright licenses. The thesis template has been adapted from Rob Robinson [168], who made his template available on his Github https://github.com/mlnotebook/thesis_template.

Copyright

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

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

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

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

Acknowledgements

I would like to extend my heartfelt thanks to Professor Ben Glocker for his exceptional mentorship and generosity throughout my PhD journey. His patience as I deepened my understanding of machine learning and his dedication during our meetings have been invaluable in building my technical foundation. I am especially grateful for his hands-on guidance, humility, and the wealth of knowledge he so generously shared. It has always amazed me how he managed to make time for all his students; his responsiveness, attentiveness, and genuine care are nothing short of phenomenal, and I deeply appreciate the support and clarity he consistently provided. I feel immensely privileged to have been supervised by him.

I am forever indebted to Professor Sir Magdi Yacoub for his unwavering support, encouragement, and mentorship over the past decade. His belief in me from the very beginning made this journey possible. I am truly grateful for how he has pushed me to grow both as a researcher and as a person. His insights have profoundly shaped my understanding of cardiovascular medicine. He has taught me to translate my research to clinical reality with both scientific curiosity and compassion. I always come away from our discussions inspired and full of new ideas. It has been a profound honour to work with and learn from such an inspirational and compassionate leader, someone I will always look up to.

Dr. Heba Aguib has been both an inspiring mentor and a true friend. I am particularly thankful for her trust in me, for championing my ideas and opening doors for me, including the opportunity to pursue this PhD. Her presence has had a lasting impact on both my academic and personal journey. My sincere thanks to Dr. Su-Lin Lee, who sparked my early interest in machine learning during my Master's studies.

I would also like to thank Dr. Wenjia Bai and Professor Daniel Rueckert for assessing my PhD milestones, I truly appreciated their time and the thoughtful comments they shared during these meetings.

To everyone in BioMedIA [1], especially those in Room 344 — thank you for making

[1] Raj, Avinash, Carolina, Mel, Sarah, big Mike, Charlie, Alex, Tian, Fiona, Marco, Fabs, Raghav, Ainkaran, Omar, Theo, Maggie, Fadi, Filippo, Wulfie, Nicole, Brian, Amelia, Carlos, Naman, Matthew, James, Kostas, Stefan, Mobarak, Zeju, Liu, Weitong, Osama, Moritz, Gavin, Miguel, Margherita, Margarita, Vassilis, Amir, Cheng, Matt, Ian and from 418/pizza Friday/GTAing: Guil, Francesco, Jess, Xiang, Denis & Mingxue.

the lab not only an inspiring place to work, but also a space filled with laughter, Mario Kart losses, puzzle frustrations (only with Avinash), chats, shared travel stories, and snacks from around the world. Your knowledge, openness, generosity, and emotional support have meant more than I can say. A very special thanks to my lifesaver Amani, for making the impossible always possible!

I am also grateful to Dr. Anees Kazi, whom I met through the MICCAI AFRICAI Summer School in 2023 — our regular meetings and collaborations have been a valuable learning experience, and I truly appreciate her insight and support.

From the Aswan Heart Centre, I would like to thank the radiology department for acquiring additional images to expand the E-HVols dataset and strengthen the impact of this research. I am also grateful to my teammates, Amr and Nagy, from the Biomedical Engineering & Innovation Laboratory (BMEIL) for their help with data annotation. I also thank Sultan, Malak, Hussam, Nadine and Omayma for their consistent support, whether with work, the PhD, or life in general. Special thanks to Mona, Alaa, Mariam, and Mohamed Salah from the Life Sciences team for helping me extract the relevant E-HVols metadata and to Bastawi from the IT department, Khaled and Abdullah from BMEIL for helping with the data transfer.

A huge thank you to my friends Marta, Juana, Eimear, and Malak for kindly proofreading this thesis. I feel incredibly lucky to have been surrounded by amazing friends throughout this journey, those who visited from abroad or stayed connected through regular calls and messages despite different time zones and busy schedules, and those I met in London who made this experience unforgettable and helped me feel truly at home. I am also grateful to London for being such a vibrant, welcoming city and for introducing me to my capoeira family and Mestre Poncianinho, whose energy and community spirit were a constant source of joy and balance. Moreover, I would like to thank Imperial College London for providing an inspiring and supportive environment to pursue my PhD, it has been a privilege to be part of such a world-leading institution.

To my family (daddy, mama, Padmini, Kitty, Soud, Youssef, Redalla, Asya, Khadija and Mariam), I love you to the moon and back. Thank you for your endless love, for planning the perfect getaways that revived me when I needed it most, and for cheering me on through every high and low. Your support and presence on countless video calls have meant the world to me. I couldn't have done this without you.

Last but not least, I am grateful to the Al-Alfi Foundation and Magdi Yacoub Foundation for generously funding my PhD.

Abstract

Graph Neural Networks (GNNs) have emerged as a powerful tool for analysing non-Euclidean data, such as meshes, which are crucial in medical imaging for modelling anatomical structures. This thesis explores the application of GNNs in shape classification tasks across neuroimaging and cardiac imaging domains, emphasising their potential for computer-aided diagnosis and disease detection. A comparative analysis of convolutional layers and node features demonstrates that Fast Point Feature Histograms (FPFH), a pose-invariant descriptor, significantly enhance GNN performance and generalisation to out-of-distribution data, simplifying preprocessing pipelines by eliminating the need for mesh alignment.

In neuroimaging, biological sex classification serves as a proof-of-concept task, with findings extended to Alzheimer's disease classification. Further investigations highlight the importance of model inspection to understand feature embeddings and encoded biases, revealing that test accuracy alone is insufficient for model selection. The fusion of imaging and shape features extracted from MRI improves predictive accuracy for clinically relevant tasks such as brain age regression and Alzheimer's disease classification. Explainability frameworks integrating multi-level insights provide transparency in model decision-making, addressing limitations of existing methods. However, these findings highlight the need for further clinical validation.

The thesis also extends GNN applications to cardiac imaging by automating aortic segmentation and mesh processing pipelines. A proposed deep learning-based registration approach outperforms traditional methods, ensuring high-quality vertex correspondence across meshes while preserving geometric features. Statistical shape analysis using Principal Component Analysis (PCA) quantifies healthy aortic shape variability and identifies correlations between shape modes and demographic or functional parameters.

Overall, this work establishes robust methodologies for leveraging GNNs in medical imaging, offering insights into architectural choices, preprocessing strategies, and interpretability frameworks that advance both neuroimaging and cardiac imaging applications.

Contents

1 Introduction **20**

1.1 Motivation . . . 21

1.2 Research aims and thesis outline . . . 23

1.3 Contributions . . . 25

1.4 List of publications . . . 26

2 Background **27**

2.1 Clinical Background . . . 28

2.1.1 Neuroimaging modalities . . . 28

2.1.2 Subcortical brain structures . . . 29

2.1.3 Cardiovascular Imaging Modalities . . . 30

2.1.4 Segments and branches of the thoracic aorta . . . 31

2.2 Common Tasks . . . 33

2.2.1 Feature extraction . . . 33

2.2.2 Dimensionality reduction . . . 33

2.2.3 Segmentation . . . 33

2.2.4 Supervised Learning . . . 33

2.2.5 Model Selection . . . 34

2.2.6 Unsupervised Learning . . . 34

2.3 Image segmentation . . . 35

2.3.1 U-Net . . . 35

2.3.2 nn-UNet . . . 35

2.3.3 TotalSegmentator . . . 36

2.3.4 Segment Anything Model (SAM) . . . 37

2.3.5 Medical Segment Anything Model (MedSAM) . . . 37

2.3.6 SAM2 . . . 37

2.3.7 MedSAM2 . . . 38

2.4 3D Data representation . . . 40

2.4.1 Euclidean . . . 40

2.4.2 Non-Euclidean . . . 43

2.5 Geometric Deep Learning . . . 44

2.5.1 Point Clouds . . . 44

2.5.2 Meshes and Graphs . . . 46
2.5.3 Graph Neural Networks . . . 47
2.6 Mesh Registration . . . 53
2.6.1 Rigid Registration . . . 54
2.6.2 Non-Rigid Registration . . . 55
2.6.3 Deformable Registration . . . 56
2.6.4 Deep Learning Approaches . . . 57
2.7 Evaluation metrics . . . 60

3 Datasets **63**
3.1 Image acquisition and meta-data . . . 64
3.1.1 UK Biobank (UKBB) . . . 64
3.1.2 Cambridge Centre for Ageing and Neuroscience (Cam-CAN) . . . 64
3.1.3 Information eXtraction from Images (IXI) . . . 64
3.1.4 Open Access Series of Imaging Studies (OASIS-3) . . . 64
3.1.5 Egyptian Healthy Volunteers (E-HVols) . . . 65
3.2 Image post-processing . . . 66
3.3 Graph representation . . . 67

4 A Comparative Study of Graph Neural Networks for Shape Classification in Neuroimaging **70**
4.1 Introduction . . . 71
4.2 Mutli-graph Architecture . . . 72
4.2.1 Selected graph convolutional operators . . . 72
4.2.2 Shared versus non-shared submodel . . . 74
4.3 Experiments . . . 76
4.3.1 Implementation and training . . . 76
4.3.2 Task 1: Biological sex classification . . . 77
4.3.3 Task 2: Alzheimer's disease classification . . . 78
4.3.4 Bias analyses . . . 78
4.4 Conclusion . . . 81

5 The Importance of Model Inspection for Better Understanding Performance Characteristics of Graph Neural Networks **82**
5.1 Introduction . . . 83
5.2 Methods . . . 84
5.2.1 Multi-graph neural network architecture . . . 84
5.2.2 Model inspection . . . 85
5.3 Experiments & Results . . . 86

5.3.1 Classification performance . . . 86
5.3.2 Effect of using structure-specific submodels . . . 87
5.3.3 Effect of mesh registration . . . 88
5.4 Conclusion . . . 90

6 Combining Imaging and Shape Features for Prediction Tasks of Alzheimer's Disease Classification and Brain Age Regression 91
6.1 Introduction . . . 92
6.2 Imaging datasets and brain shape representation . . . 94
6.3 Feature extraction & fusion . . . 95
6.4 Experiments & Results . . . 96
6.4.1 Brain age regression . . . 96
6.4.2 Alzheimer's disease classification . . . 97
6.5 Discussion . . . 99
6.6 Conclusion . . . 100

7 Limitations of GNN explainability methods: A case study on sex classification from brain shapes 101
7.1 Introduction . . . 102
7.2 Methods . . . 104
7.2.1 Model and datasets . . . 104
7.2.2 Substructure perturbation and model calibration . . . 104
7.2.3 Complementary evaluations of importance scores . . . 105
7.2.4 Shape variation . . . 107
7.3 Results . . . 108
7.3.1 Model calibration . . . 108
7.3.2 Ranking using substructure perturbation . . . 108
7.3.3 Analysing GNNExplainer importance scores . . . 109
7.3.4 Label-specific shape variation . . . 111
7.4 Conclusion . . . 116

8 Automated Segmentation of the Aorta - E-HVols 117
8.1 Introduction . . . 119
8.2 Automated segmentation . . . 120
8.2.1 Image pre-processing . . . 121
8.2.2 Model selection for segmentation inference . . . 121
8.2.3 Quality control . . . 122
8.3 Results . . . 123
8.3.1 TotalSegmentator . . . 123
8.3.2 MedSAM2 . . . 123

8.3.3 Fine-tuned TotalSegmentator . . . 123
8.3.4 nn-UNet . . . 124
8.3.5 Failed segmentation by nn-UNet - test data . . . 124
8.4 Conclusion . . . 126

9 Aortic Mesh Processing and Atlas Creation 127
9.1 Introduction . . . 128
9.2 3D surface reconstruction . . . 131
9.2.1 Iso-surface Extraction . . . 131
9.2.2 Post-processing . . . 131
9.3 Alignment and Correspondence . . . 133
9.4 Identifying aortic shape variations . . . 137
9.5 Results . . . 138
9.5.1 Mesh registration . . . 138
9.5.2 Shape analysis using PCA . . . 138
9.6 Conclusion . . . 143

10 Characterising Aortic Shape Variability 144
10.1 Introduction . . . 145
10.2 Methods . . . 146
10.2.1 Imaging dataset and pre-processing . . . 146
10.2.2 Target variables . . . 146
10.2.3 Data exploration . . . 149
10.2.4 Power analysis . . . 150
10.2.5 Linear regression & baselines . . . 150
10.2.6 Single and multi-target prediction using our GNN . . . 150
10.3 Experiments . . . 151
10.4 Results . . . 152
10.4.1 Power analysis . . . 152
10.4.2 Target variables and PCA projections relationships . . . 152
10.4.3 Linear regression models . . . 155
10.4.4 Effect of mesh processing on GNN performance . . . 155
10.4.5 GNN performance predicting selected target variables . . . 155
10.5 Conclusion . . . 160

11 Conclusions 161
11.1 Main conclusions . . . 162
11.2 Limitations . . . 164
11.3 Future work . . . 166

References **167**

Appendices **181**

List of Figures

2.1 (a) A slice of a T1-weighted MR stack showing the 15 subcortical brain structures segmented. (b) The corresponding extracted shapes as triangular meshes. . . . 29
2.2 Schematic of segments and branches of the aorta. . . . 32
2.3 U-Net architecture [172] . . . 36
2.4 SAM overview [110]: . . . 37
2.5 MedSAM [133] . . . 38
2.6 SAM2 overview [162]: . . . 38
2.7 MedSAM2 [134] . . . 39
2.8 3D Data Representations [2] . . . 40
2.9 The two main categories of 3D descriptors: signatures and histograms [176]. . . . 42
2.10 Illustration of a graph-based network [77] . . . 43
2.11 GNN model pipeline [245] . . . 47
2.12 Types of GNNs [222] . . . 48
2.13 Taxonomy of point cloud rigid registration methods [132] . . . 58
2.14 Taxonomy of deep learning based Point Cloud Registration methods, denoted as PCR in the figure [238] . . . 59
2.15 AUROC . . . 62

3.1 PFH calculation . . . 67
3.2 FPFH . . . 69

4.1 Proposed multi-graph architecture . . . 71
4.2 Non-shared submodel (left), shared submodel (right) . . . 75
4.3 Aspects of comparison; choice of node features (left), data augmentation by perturbing the nodes (top-right) and type of convolutional layers (bottom-right) . . . 76
4.4 ROC curves for sex classification . . . 78
4.5 ROC curves . . . 79
4.6 Effect of convolution layer . . . 80
4.7 Bias analysis for sex classification . . . 80

5.1 Model architecture . . . 84

5.2 Sex classification performance 87
5.3 Effect of modelling choices on feature separability 89

6.1 **Fusion model overview**, consisting of image-based and graph-based feature extractors, embeddings concatenated and passed through a multi-layer perceptron (MLP). 95
6.2 Scatter plots of true age against predicted age 97
6.3 AD classification results from using embeddings from baseline models, passing them through the MLP (one FC). 98

7.1 Perturbing the R_Puta 105
7.2 Generating the brainstem box plot using both node importance distribution methods by averaging across subjects (left) and per substructure (right). Crossed-out node mask values are not part of the top-k values. 106
7.3 Model Calibration 108
7.4 Difference in probabilities before and after perturbation. 109
7.5 Model Explainability (per substructure) 110
7.6 Phenomenon Explainability (per substructure) 111
7.7 Model Explainability (across subjects) 112
7.8 Phenomenon Explainability (across subjects) 113
7.9 The anterior and posterior views are shown on the left and right, respectively, for each configuration (model or phenomenon), explainability and each subgroup. The top row is for females (a) and (b)), the middle row is for males (c) and (d), and the bottom row is for both males and females (e) and (f). The importance scores are colour-coded (low blue, high red) and superimposed on the average shapes. 114
7.10 Top PCA modes and the difference between mean males and females. Please note that the colour bars have different ranges so that the variability is visible. 115

8.1 Using Mimics for 3D reconstruction of the aorta. 120
8.2 Segmentation results. 123
8.3 Failed segmentation 125

9.1 Overview for the developed pipeline 130
9.2 Overview for 3D surface reconstruction steps following segmentation. . 131
9.3 Template selection, alignment and registration. 134
9.4 Comparison of rigid and non-rigid registration methods using different evaluation metrics 139

9.5 A random target mesh was selected to visualise how the template deformed . . . 140
9.6 Percentage of variance explained . . . 140
9.7 The panels from left to right show the average shape in bright red, followed by the top 6 modes of variation. . . . 142

10.1 Slice from short-axis stack of the LV. . . . 147
10.2 Magnitude and phase images from the reference scan (top row) and velocity-sensitive scan (bottom row). Subtracting the phase images gives the velocity map [63]. . . . 148
10.3 Size-related parameters. . . . 149
10.4 Power Curves for Multiple Linear Regression. . . . 152
10.5 Comparison of correlations between PCA modes and target variables after applying different Umeyama methods. . . . 154
10.6 Sex classification performance: (a) using SplineConv for post-processed registered meshes (Reg) vs meshes produced by Marching Cubes (MC) (b) using the meshes that came out of MC with GCNConv vs SplineConv 158

List of Tables

2.1 Spectral vs Spatial methods [222] 51

6.1 Evaluation metrics for brain age regression using embeddings extracted from image- and graph-based models. 96
6.2 AD classification results 97

9.1 **Variance explained by the top 10 principal components** for aortic shape analysis after applying Umeyama rigid and similarity transformations separately. 141

10.1 Target variables showing strong correlation ($|r| > 0.2$ in $\geq$2 PCA modes) with shape descriptors derived from rigid and similarity registrations. . . 153
10.2 MAE and R^2 for regression models using different combinations of PCA modes after **rigid Umeyama** to predict available target variables 156
10.3 MAE and R^2 for regression models using different combinations of PCA modes after **similarity Umeyama** to predict available target variables . 157
10.4 Performance of single-target GNN models 158
10.5 Performance of four multi-target GNN models 159

Chapter 1

Introduction

This thesis begins by introducing the foundational concepts and motivations driving this research: Chapter 1 and Chapter 2. Then, in Chapter 3, we list the imaging datasets used in the studies presented in the succeeding chapters. In Chapter 4, we explore the application of graph neural networks (GNNs) in medical imaging, starting with a comparative study to investigate how different design choices impact model performance and generalisation, providing key insights for practitioners. Chapter 5 extends this analysis by emphasising the importance of model inspection, specifically examining the effect of graph convolutional submodels and mesh registration on feature learning in brain shape classification. Building on this, Chapter 6 explores the fusion of shape and imaging features for brain age prediction and Alzheimer's classification, demonstrating improved prediction accuracy by combining shape embeddings from a GNN with appearance features from Magnetic Resonance (MR) scans. Chapter 7 delves into model explainability to understand which parts of the input graph influence model decisions. In Chapter 8, we address the challenges of applying GNNs to aortic MR images by developing a robust segmentation and mesh reconstruction pipeline. Finally, Chapter 9 presents an advanced deep learning approach for mesh registration, demonstrating its superiority in establishing accurate correspondence across subjects, and providing a foundation for morphological studies, followed by the shape analysis of the aorta, which is presented in Chapter 10.

1.1 Motivation

The rapid advancement of medical imaging technologies has revolutionised our ability to visualise and analyse complex anatomical structures. This progress has opened new avenues for understanding human physiology, diagnosing diseases, and developing personalised treatment strategies [153]. However, the increasing complexity and volume of medical imaging data present significant challenges in terms of interpretation and analysis [119]. Traditional methods of image analysis often fall short when dealing with the intricate, three-dimensional nature of anatomical structures, particularly when attempting to capture subtle variations that may be indicative of disease or individual differences [221].

In recent years, deep learning techniques have shown remarkable success in various domains of medical image analysis, from segmentation to classification tasks [196]. Among these techniques, Graph Neural Networks (GNNs) have emerged as a particularly promising approach for analysing non-Euclidean data such as 3D shapes as meshes, which are commonly used to represent anatomical structures. GNNs offer a unique ability to capture both local and global structural information, making them well-suited for tasks involving complex geometries and topologies [106, 117].

Despite their potential, the application of GNNs in medical imaging, particularly in the analysis of anatomical shapes, is still in its infancy. There is a pressing need for comprehensive studies that explore the effectiveness of different GNN architectures, node features, and training strategies in the context of medical shape analysis. Moreover, the interpretability of these models remains a significant challenge, hindering their widespread adoption in clinical settings where transparency and explainability are crucial.

This thesis is motivated by the need to bridge these gaps in knowledge and methodology. By conducting extensive comparative analyses, investigating fusion approaches, and addressing interpretability challenges, we aim to advance GNN-based medical shape analysis. Our work spans both neuroimaging and cardiac imaging domains, demonstrating the versatility and potential of these techniques across different anatomical structures and clinical applications.

Furthermore, we are driven by the potential clinical impact of these advanced analytical methods. Improved shape analysis techniques could lead to earlier and more accurate disease detection, better understanding of anatomical variations in healthy populations, and more personalised treatment planning.

Through this research, we hope to not only advance the technical aspects of medical

image analysis but also to pave the way for more widespread adoption of these techniques in clinical practice. By addressing key challenges in the field, from architectural choices to model explainability, we aim to increase the confidence of clinicians and researchers in using GNN-based approaches for medical shape analysis, ultimately contributing to improved healthcare outcomes.

1.2 Research aims and thesis outline

This research utilises several large-scale neuroimaging datasets, including UKBB, CamCAN, IXI, and OASIS3, as well as the E-HVols, a cardiac dataset from the Aswan Heart Centre (AHC). These diverse datasets allow for comprehensive analysis across various anatomical structures and clinical conditions, allowing testing for the generalisation and robustness of our methods. A detailed description of these datasets is presented in Chapter 3.

Despite the growing popularity of GNNs in medical image analysis, comprehensive studies comparing different GNN architectures and methodologies in this context are lacking. Our comparative study from Chapter 4 provides valuable insights into the most effective approaches for shape classification and regression. This study aims to equip practitioners with a deeper understanding of the current state-of-the-art approaches for applying GNNs to medical shape analyses.

Understanding the inner workings of GNNs is crucial for their practical application in medical contexts. This is particularly important when using the extracted features for downstream tasks such as fine-tuning, domain transfer, or multi-modal approaches. Therefore, in Chapter 5, we investigate the impact of modelling choices on feature learning characteristics, emphasising the importance of looking beyond test accuracy to identify potential biases and non-discriminative features.

While exploring the evolution of Convolutional Neural Networks (CNNs) and GNNs, it has become apparent that combining graph and image-based features would build on both strengths. Most fusion models combine different imaging modalities, but combining graph and image-based features from the same modality has not been explored. We investigate the potential of combining features extracted from ResNet and our developed graph-based model in Chapter 6. This fusion approach aims to enhance the performance of either model alone.

A significant limitation of GNNs, like many deep learning models, is their lack of interpretability. Chapter 7 addresses this limitation by comparing different explainability configurations to provide insights into the decision-making process of our models. By enhancing model transparency and interpretability, we aim to uncover potential phenotypical biomarkers that could provide novel insights to clinicians.

To extend the application of our methods to cardiac imaging, Chapter 8 focuses on automating aortic segmentation from MR images using the E-HVols dataset. We develop a custom data pre-processing pipeline that emphasises maintaining consistent image orientation across the entire dataset.

Chapter 9 presents a comprehensive pipeline for processing and analysing aortic meshes extracted from medical imaging segmentation. We compare various registration methods, including traditional approaches and a proposed deep learning method, to establish accurate vertex correspondence across meshes.

In Chapter 10, we conduct a comprehensive analysis of aortic shapes reconstructed from MR images. We use Principal Component Analysis (PCA) to quantify healthy shape variability and investigate correlations between shape modes and demographic or functional parameters. We also compare the predictive performance of linear regression models using PCA modes to our bespoke GNN model.

This thesis concludes with a summary of our findings, their implications for medical image analysis, limitations and potential future research directions in Chapter 11.

1.3 Contributions

Multi-Graph Architecture Model: Designed a model to utilise multiple structures as subgraphs. We compared two configurations: a shared submodel that learns generalised representations from all subgraphs, and a non-shared submodel, where each subgraph has its own submodel learning subgraph-specific representations.

Model Inspection Pipeline: Developed a pipeline to demonstrate the importance of identifying which model choices extract the best discriminative features despite overall model performance. This identification is critical when using extracted features for downstream tasks.

Fusion Model: Fusion models often use different imaging modalities but not different data representations. Therefore, we investigated a fusion model combining image-based and graph-based features.

Model Explainability Methods: We introduce a calibration-aware, perturbation-based framework for GNN explainability in our multi-graph model. By integrating subgraph-level perturbations, our method captures both local and global dependencies. We also analysed node importance scores generated by GNNExplainer across-subject consistency to identify population-level biomarkers and within-subgraph critical nodes to pinpoint localised drivers. Both methods yield complementary results, underscoring the need for clinical validation.

Comprehensive Shape Analysis Framework:

- Compared state-of-the-art segmentation models for segmenting the thoracic aorta from MRI.
- Developed a custom pipeline for surface reconstruction from the segmented images while maintaining a fixed number of nodes and establishing correspondence across all aortic shapes in the dataset.
- Constructed an Atlas and identified key shape variations using Principal Component Analysis (PCA).

1.4 List of publications

- **Shehata, N.**, Bain, W., and Glocker, B. “A Comparative Study of Graph Neural Networks for Shape Classification in Neuroimaging”. In: Proceedings of the First International Workshop on Geometric Deep Learning in Medical Image Analysis. Vol. 194. Proceedings of Machine Learning Research. PMLR, 18 Nov 2022, pp. 160–171 proceedings.mlr.press/v194/shehata22a.html
- Inácio MH, Shah M, Jafari M, **Shehata N**, Meng Q, Bai W, Gandy A, Glocker B, O’Regan DP. “Cardiac age prediction using graph neural networks”. MedRxiv. 2023 Apr 20:2023-04.
- **Shehata, N.**, Pic¸arra, C., Kazi, A., et al. “The Importance of Model Inspection for Better Understanding Performance Characteristics of Graph Neural Networks”. In: 2024 IEEE International Symposium on Biomedical Imaging (ISBI). 2024, pp. 1–5. DOI: 10.1109/ISBI56570.2024.10635526.
- **Shehata N**, Piçarra C, Glocker B. “Combining imaging and shape features for prediction tasks of Alzheimer’s disease classification and brain age regression”. arXiv preprint arXiv:2501.07994. 2025 Jan 14 - presented as an oral presentation at UK AI conference.
- **Shehata N**, Elsawy A, Nagy M, ElMahdy M, Ali M, Romeih S, Aguib H, Yacoub M, Glocker B. “A Comprehensive Pipeline for Aortic Segmentation and Shape Analysis” - accepted for STACOM 2025.

Chapter 2

Background

This chapter presents the foundation for the methods we developed, which will be presented in the next chapters. Under the clinical background, section 2.1, we first present common imaging techniques and what information can be extracted from these different imaging modalities for clinical assessment or diagnosis. A simple description of the anatomical structures we will later study throughout this thesis is also presented in this section. Section 2.2 gives an overview of common tasks that can be performed on the extracted data from images. Section 2.3 lists the current state-of-the-art models used for medical image segmentation and how to represent the segmented data in a non-Euclidean space, as explained in section 2.4. Geometric deep learning approaches to analyse data in non-Euclidean space and their applications towards the aforementioned tasks are presented in section 2.5. In section 2.6, we give some background on mesh registration. Mesh registration is an optional step commonly used as a pre-processing step before using the extracted shapes in further analyses. Finally, section 2.7 lists the common evaluation metrics for tasks, including segmentation, registration, classification and regression.

2.1 Clinical Background

In this section, we give some neuroimaging and cardiovascular imaging background because, as discussed in Chapter 3, more readily available datasets are available for brain subcortical structures and less for the thoracic aorta. We, therefore, developed and evaluated our methods using the brain datasets and then extended their applications to the aorta.

2.1.1 Neuroimaging modalities

Neuroimaging encompasses a variety of advanced imaging techniques that provide critical insights into the structure and function of the brain. These modalities are indispensable tools in both clinical and research settings, enabling the diagnosis, monitoring, and understanding of neurological conditions.

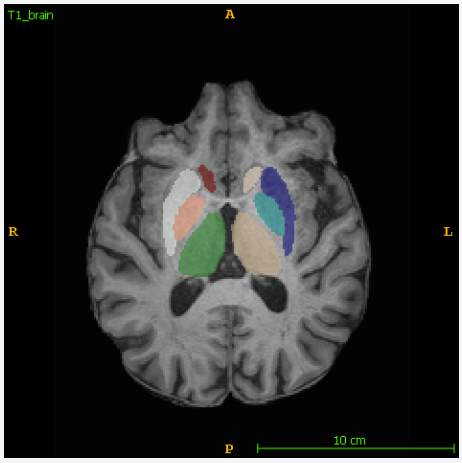
- **Magnetic Resonance Imaging (MRI) and Functional MRI (fMRI):** MRI is a non-invasive imaging technique that uses magnetic fields and radio waves to produce detailed images of brain anatomy. It is widely used for detecting structural abnormalities, such as tumours or lesions. Functional MRI (fMRI), on the other hand, measures brain activity by detecting changes in blood flow, offering insights into brain functions and connectivity. fMRI is particularly useful for pre-surgical planning and understanding brain activity related to cognitive tasks
- **Computerized Tomography (CT):** CT scans use X-rays to create cross-sectional brain images. They are often used in emergencies to assess acute conditions like haemorrhages or fractures. CT is valued for its speed and ability to provide detailed images of bone and calcifications
- **Positron Emission Tomography (PET):** PET scans involve using radioactive tracers to visualise metabolic processes in the brain. This modality is particularly useful for detecting areas of abnormal metabolic activity, such as epilepsy or Alzheimer's disease. PET can assess brain function and detect tumours by highlighting regions with altered glucose metabolism
- **Electroencephalography (EEG) and Magnetoencephalography (MEG):** EEG measures electrical activity in the brain using electrodes placed on the scalp, while MEG detects magnetic fields produced by neuronal activity. Both techniques are crucial for evaluating conditions like epilepsy and sleep disorders. MEG, in particular, provides precise localisation of brain activity, which is beneficial for surgical planning
- **Functional Near-Infrared Spectroscopy (fNIRS):** fNIRS is a non-invasive method

that measures brain activity by detecting changes in blood oxygenation. It is often used in cognitive and developmental neuroscience research due to its portability and ability to monitor brain function in naturalistic settings

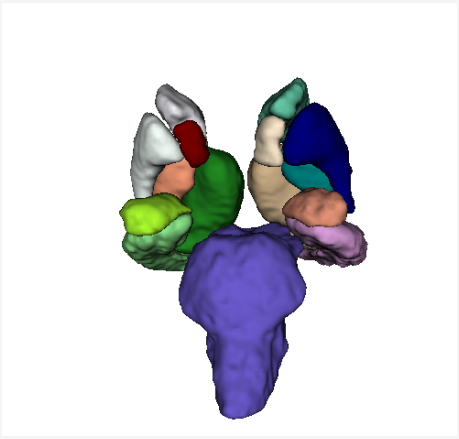
These neuroimaging modalities are integral to modern neurology and psychiatry, offering a comprehensive view of brain structure and function. They facilitate the diagnosis of neurological disorders, guide treatment planning, and enhance our understanding of brain-behaviour relationships. By integrating data from multiple imaging techniques, clinicians and researchers can obtain a more holistic view of brain health, ultimately improving patient outcomes.

2.1.2 Subcortical brain structures

The human brain contains several crucial subcortical structures that play vital roles in various cognitive, emotional, and motor functions. The subcortical structures below work in concert, forming intricate networks that underpin complex human behaviours and cognitive processes, as shown in Fig.2.1.



(a) Segmented subcortical brain structures



(b) Extracted subcortical brain shapes

Figure 2.1: (a) A slice of a T1-weighted MR stack showing the 15 subcortical brain structures segmented. (b) The corresponding extracted shapes as triangular meshes.

The **brain stem**, located at the base of the brain, is responsible for regulating essential life functions such as breathing, heart rate, and blood pressure. The **left and right thalami** act as relay centres, processing and integrating sensory and motor information before transmitting it to the cerebral cortex. The basal ganglia include the **caudate nucleus and putamen**, collectively known as the striatum, involved in motor control, learning, and cognitive functions and the **pallidum**, which plays a crucial role in regulating voluntary movement. The temporal lobe includes the **hippocampus**, a

seahorse-shaped structure essential for forming new memories and spatial navigation and the **amygdala**, an almond-shaped structure primarily responsible for processing emotions, particularly fear and aggression. Lastly, the **accumbens** area, often called the nucleus accumbens, is part of the brain's reward circuit and involves motivation, pleasure, and addiction-related behaviours.

2.1.3 Cardiovascular Imaging Modalities

Cardiovascular imaging encompasses a range of advanced modalities essential for diagnosing, managing, and understanding heart diseases. These techniques provide detailed insights into the heart's structure and function, offering invaluable information for clinical decision-making.

Medical imaging is indispensable for diagnosing patients with aortic diseases. Various imaging modalities are used in the diagnosis, postoperative, and follow-up phases. Each modality has its advantages and limitations, which are reviewed in this section. Other modalities, like echocardiography, duplex, and intravascular ultrasound, are not reviewed but are also used.

- **Echocardiography:** This modality uses ultrasound waves to create moving images of the heart. It is widely used because it provides comprehensive data without exposing patients to radiation. Transthoracic echocardiography (TTE) is the most common type, allowing assessment of heart size, shape, wall thickness, and valve function. Stress echocardiography evaluates heart performance under stress, helping diagnose conditions like coronary artery disease.

- **Cardiac Magnetic Resonance Imaging (MRI):** Cardiac MRI uses magnetic fields and radio waves to produce detailed images of the heart's anatomy and function. It is particularly useful for assessing myocardial tissue composition, detecting cardiac tumours, and evaluating congenital heart defects. MRI provides high-resolution images and can assess myocardial viability and perfusion without ionising radiation. Accurate aortic measurements can be obtained from 3D steady-state free precision MRI of the aorta, which matches well with those from TTE [126]. Another commonly used technique to calculate through-plane or in-plane blood velocity is two-dimensional phase contrast MRI, which encodes velocities in the phase images' pixel intensities. An extension to that is 3D velocity encoding synchronised with ECG signal, named 4D flow MRI, providing time-resolved, 3D velocity fields [50].
  In comparison to CT, longer acquisition times are needed, and some difficulty is present when monitoring unstable subjects. MRI is a good option for follow-up studies in stable patients with known aortic disease since it does not require ion-

ising radiation or iodinated contrast [15].

- **Computed Tomography (CT):** Cardiac CT, including coronary CT angiography (CCTA), combines X-ray images to create detailed cross-sectional views of the heart and its vessels. It is a non-invasive method for evaluating coronary artery disease and structural heart issues. CT calcium scoring assesses the risk of heart disease by detecting calcified plaques in coronary arteries. Multi-detector computed tomography (MD-CT) reduces image acquisition time and produces images with enhanced spatial and temporal resolution compared to single-detector CT. The main limitation is that the subject needs to be injected with a contrast agent so that the images capture the agent's arterial transit [143]. Capturing electrographic (ECG)-gated CT images is another technique of acquiring CT images, allowing more accurate measurements of the aortic segments and reducing artefacts related to cardiac motion [205]. Another compromising factor is the exposure to ionising radiation, although there are ways to reduce it, but not eliminate it [44].
- **Nuclear Imaging:** This includes myocardial perfusion imaging using radioactive tracers to assess blood flow to the heart muscle. It is valuable for evaluating coronary artery disease and myocardial viability. Techniques like SPECT and PET provide functional information about the heart, complementing anatomical data from other modalities.
- **Coronary Angiography:** Although invasive, coronary angiography remains the gold standard for visualising coronary arteries. It involves using a catheter to inject contrast dye into the coronary vessels, providing detailed images to guide interventions such as angioplasty.

These cardiovascular imaging modalities are integral to modern cardiology, allowing for early detection, accurate diagnosis, and effective management of heart conditions. By integrating data from multiple imaging techniques, clinicians can comprehensively understand cardiovascular health, ultimately improving patient outcomes.

2.1.4 Segments and branches of the thoracic aorta

The aorta starts at the aortic root, which gives rise to two major coronary arteries and ends by splitting into the common iliac arteries. This major vessel can be subdivided into seven segments; the aortic root, the ascending aorta, the ascending aortic arch, the vessel bearing arch, the aortic isthmus, the thoracic descending and abdominal aorta [52], Fig.2.2.

Aortic Root The complex structure of the aortic root maintains a normal function under different hemodynamic conditions. Its components' shape and size have a major role

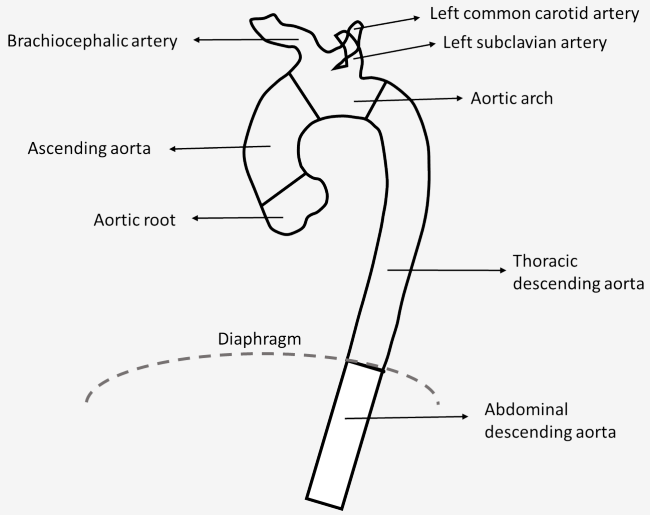


Figure 2.2: Schematic of segments and branches of the aorta.

in vortices development, leading to minimising mechanical stress on the cusps (valve leaflets) and controlling their motion throughout the cardiac cycle [226]. It consists of the annulus, sino-tubular junction (STJ) and sinuses of Valsalva. The junction between the aorta and the left ventricle is the annulus. Meanwhile, the STJ is where the structure evolves into the tubular shape of the ascending aorta. The sinuses of Valsava form the segment between the annulus and STJ. These sinuses have out pouches of the aortic wall, one associated with each aortic cusp. The three sinuses are named according to the coronary arteries originating from them; left-, right- and non-coronary sinus. The recirculation of blood in the sinuses during systole [72, 101] is affected by the bulging shape of the aortic root, which is also vital for feeding the coronaries supplying the heart [147]. The annulus forms an elliptical shape during diastole, becoming fully circular during cardiac ejection with an area larger than that of the STJ [113, 202], which creates a velocity gradient across the aortic root allowing blood to flow from a larger cross-section to a smaller one [78].

Ascending Aorta The ascending aorta starts from the aortic orifice in the left ventricle and ascends to become the **aortic arch** from which three major vessels branch out in order of proximal to distal: the brachiocephalic trunk innominate artery, the left common carotid artery and the left subclavian artery [56]. The remaining segment is then the **descending aorta**, which supplies blood to the lower body.

2.2 Common Tasks

Computational methods are usually developed to perform one or more of the tasks below as accurately as possible. This section briefly explains these tasks, which are often mentioned in the next chapters.

2.2.1 Feature extraction

Feature extraction is a process of dimensionality reduction by which the initial raw data set is reduced without losing the ability to describe the original data set accurately. It also eliminates redundant data, speeding up the learning and generalisation steps that will be followed in any machine learning process.

2.2.2 Dimensionality reduction

Dimensionality reduction transforms high-dimensional data into a lower-dimensional representation while preserving essential structures. This helps improve model efficiency and prevent overfitting [48]. Two common approaches are:

Principal Component Analysis (PCA): PCA projects data into new variables (called **principal components**) that are linear combinations of the original features, capturing the maximum variance in fewer dimensions.

Penalised Regression: Methods like ridge regression and lasso regression impose penalties on model complexity, effectively reducing the number of variables while maintaining predictive power.

2.2.3 Segmentation

Segmentation is the process of assigning each pixel a label (or belonging to a class). This is useful for applications that require high accuracy and are manually intensive because they require pixel-level accuracy. As a result, automated and semi-automated methods have been developed to reduce manual effort. When pixels are assigned a single label, the result is a mask that outlines the shape of the object in the image being segmented (the Region of Interest).

2.2.4 Supervised Learning

Classification groups data into predefined classes. The classes are often called targets, labels, or categories assigned to given data points. Classification falls under supervised learning, where the targets are available and provided with input data [135].

Regression Regression predicts a continuous variable y based on one or more predictor variables x [135].

Linear Regression: Assumes a linear relationship between the outcome and predictors. The equation is $y = b_0 + b * x + e$, where b_0 is the intercept, b is the coefficient, and e is the residual error. The Ordinary Least Squares method determines the weights to minimise prediction error.

Non-linear Regression: If the relationship is not linear, polynomial and spline regression methods can be used instead.

2.2.5 Model Selection

Model selection identifies the best combination of predictors to build an optimal predictive model. This involves comparing multiple models with different predictor sets to select the one that minimises error [241]. Common techniques include:

Best Subsets Regression: Evaluates all possible combinations of predictors to find the best-performing model.

Stepwise Regression: Iteratively adds or removes predictors based on statistical significance.

2.2.6 Unsupervised Learning

Clustering groups data points into clusters based on inherent similarities without predefined labels. This is useful for discovering patterns, reducing data complexity, and detecting anomalies [144]. Common clustering methods include:

Density-Based Methods: Identify clusters as dense regions in feature space, allowing flexible cluster shapes.

Hierarchical Methods: Form clusters in a tree-like structure, either bottom-up (Agglomerative) or top-down (Divisive).

Partitioning Methods: Divide data into clusters by optimising an objective criterion, such as minimising intra-cluster variance.

Grid-Based Methods: Organise data into a grid structure, enabling fast clustering regardless of dataset size.

2.3 Image segmentation

Image segmentation is a critical task in computer vision that involves partitioning an image into distinct regions or objects. This process is foundational in various applications, including medical imaging, robotics, and object detection. Initially, image segmentation relied on classical machine learning techniques such as clustering algorithms (e.g., K-means), support vector machines (SVMs), and random forests. These methods grouped pixels into segments based on similar features like colour and texture. While adequate for simple tasks, these techniques often struggled with complex scenes and required extensive manual feature engineering [180].

The advent of deep learning, particularly Convolutional Neural Networks (CNNs), marked a significant shift in image segmentation. CNNs automate feature extraction, allowing for more accurate and efficient segmentation. CNN-based models have achieved state-of-the-art performance in image segmentation [165]. They excel in capturing spatial hierarchies in images, making them particularly suited for tasks requiring fine-grained segmentation. We present some selected models below.

2.3.1 U-Net

U-Net is a pioneering architecture specifically designed for biomedical image segmentation. It follows an encoder-decoder structure to capture both contextual and spatial information. The encoder captures context through downsampling, and the decoder reconstructs the segmentation map through upsampling. U-Net introduced skip connections between corresponding layers in the encoder and decoder, preserving spatial information (high-resolution features) lost during downsampling, Fig. 2.3. This architecture has become a standard in medical image segmentation because it can produce precise segmentations with limited training data [172].

Over time, U-Net's two-dimensional design highlighted a limitation: ignoring valuable context along the third dimension could hinder segmentation quality for 3D medical images. To address this, Çiçek et al. expanded the U-Net's topology to include 3D convolutions, allowing the model to handle volumetric data better and further solidify its relevance for biomedical imaging [34].

2.3.2 nn-UNet

Building on U-Net's success, nnU-Net (no-new-Net) [92] extended the concept by introducing an automated, data-driven approach. Unlike U-Net, it is designed to adapt to any medical imaging dataset without manual intervention. nnU-Net analyses the properties of each dataset, such as image size, voxel spacing, and intensity distribution,

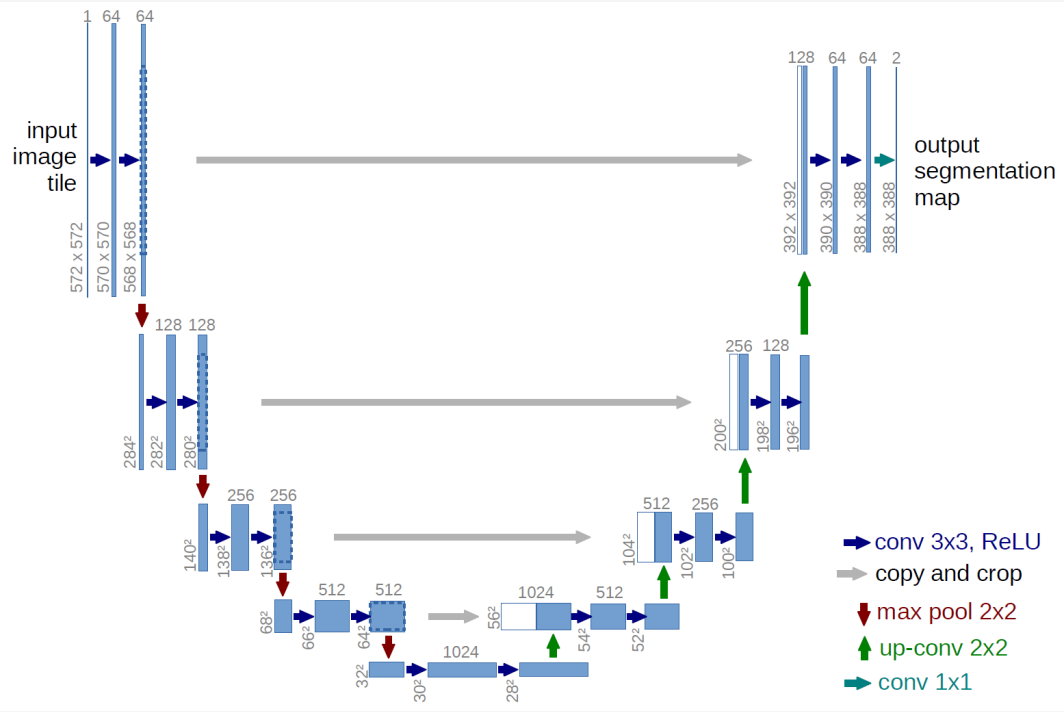


Figure 2.3: U-Net architecture [172]

to tailor preprocessing, network architecture, training schedules, and post-processing steps accordingly [93]. By dynamically configuring itself, nnU-Net eliminates manual adjustments, making it highly versatile for imaging modalities such as MRI and CT. This open-source framework has consistently achieved state-of-the-art performance across diverse segmentation challenges, including the Medical Segmentation Decathlon[93]. Moreover, a review in 2024 showed that nnU-Net's convolution-based approach often outperformed newer models, such as transformer-based architectures, solidifying its reputation as a robust and adaptable tool in medical image segmentation [94].

2.3.3 TotalSegmentator

TotalSegmentator [37] extends the capabilities of nn-UNet by incorporating multi-scale features and attention mechanisms to improve segmentation accuracy by focusing on relevant regions. In a single pass, it provides a comprehensive, full-body anatomical segmentation from medical images, including the brain, heart, lungs, liver, and kidneys. Released after nnU-Net, it leverages deep learning techniques to perform detailed segmentations while efficiently managing large-scale data, balancing high performance with computational efficiency.

2.3.4 Segment Anything Model (SAM)

SAM [110] introduces a promptable interface for segmentation, where prompts can include points, bounding boxes, or masks that guide the model in identifying regions of interest within an image, fig 2.4.

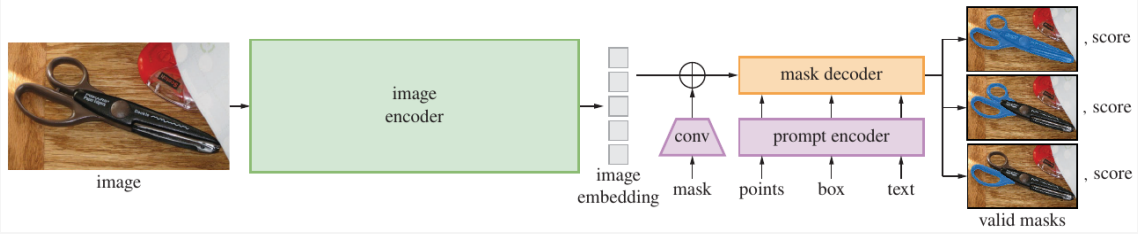


Figure 2.4: SAM overview [110]: image encoder outputs an image embedding queried by input prompts to produce object masks. SAM can output multiple valid masks and confidence scores for ambiguous prompts corresponding to more than one object.

SAM's architecture includes a Vision Transformer (ViT) backbone, which processes visual features at multiple levels, enabling robust and fast segmentation of diverse images. The success of SAM lies in its ability to generalise to numerous image types without fine-tuning, making it a powerful tool in open-world image segmentation scenarios, as it uses a dataset of over 1 billion masks and 11 million images.

2.3.5 Medical Segment Anything Model (MedSAM)

MedSAM [133] adapts SAM specifically tailored for medical image segmentation tasks, Fig. 2.5. It retains SAM's core architecture while being fine-tuned for medical imaging applications. MedSAM can be guided through the segmentation process using traditional point-and-box prompts, making it easy for medical professionals to interact. It performs well across various medical imaging modalities but may encounter challenges with highly detailed boundaries or small anatomical structures. MedSAM is trained directly on medical imaging datasets with 1,570,263 image-mask pairs, covering 10 imaging modalities. It is trained to learn the specific features and patterns common in medical images. While effective for general medical image segmentation tasks, its single-stage training approach inherits some limitations from the base SAM model, particularly when dealing with complex medical scenarios or achieving high boundary precision.

2.3.6 SAM2

SAM2 [162] builds upon SAM by introducing capabilities for handling video and dynamic image sequences, addressing the demand for real-time segmentation in appli-

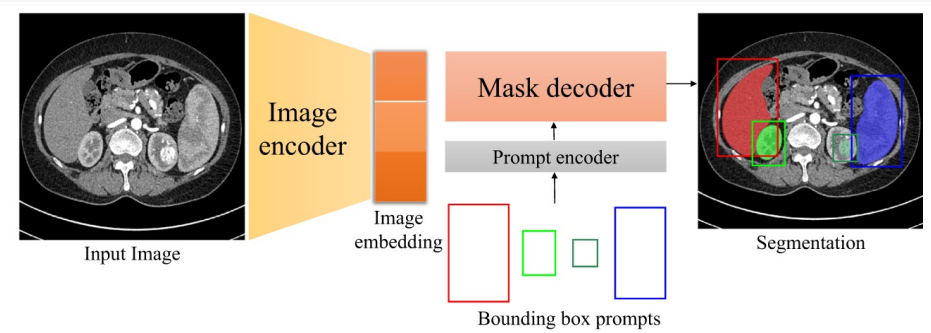


Figure 2.5: MedSAM [133]

cations like autonomous driving and video-based object tracking. In SAM2, segmentation can be guided with one prompt per sequence, allowing users to set a single segmentation prompt that the model applies across an entire video or image sequence, thus minimising the need for repetitive manual intervention. SAM2 retains the versatility of SAM but is optimised for sequence-based data, with a refined attention mechanism that tracks and adjusts segmentation targets across frames. This improvement enables SAM2 to maintain spatial and temporal consistency in segmentations, making it particularly effective for applications that rely on tracking objects or features through time-lapse or live-feed data 2.6.

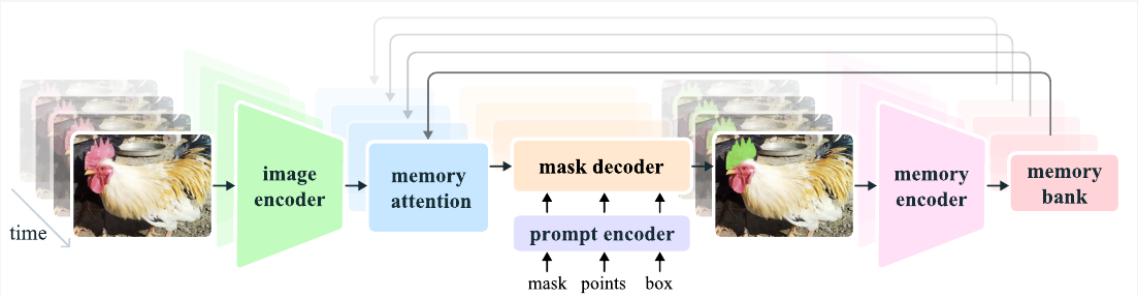


Figure 2.6: SAM2 overview [162]: Segmentation prediction for each frame is based on the current prompt and previously observed memories. Videos are processed in a streaming manner, with the image encoder analysing one frame at a time while referencing memories from past frames of the target object. The mask decoder, which may also use input prompts, generates the segmentation mask for the current frame. Finally, a memory encoder converts the prediction and image encoder embeddings (not shown in the figure) for future frame use.

2.3.7 MedSAM2

MedSAM2 [134] further extends the capabilities of SAM and MedSAM to cover 3D imaging and time-series medical data, such as ultrasound scans and real-time monitoring. Key features include “One-Prompt Segmentation,” enabling a single prompt to

be applied consistently across multiple frames or slices, which is particularly beneficial for monitoring anatomical changes over time, fig. 2.7.

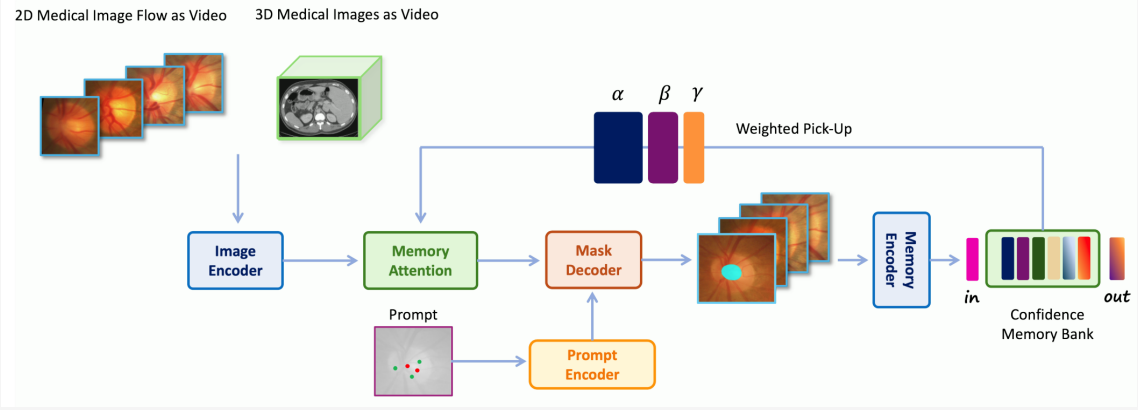


Figure 2.7: MedSAM2 [134]

This model also employs self-supervised learning and domain adaptation for improved performance across various imaging modalities and datasets. Its architecture includes fine-tuning the image encoder and mask decoder while freezing the prompt encoder to prevent overfitting, alongside advanced preprocessing and data augmentation techniques to accommodate the variability in medical images.

Additionally, MedSAM2 introduces a sophisticated two-stage training strategy that markedly improves its performance over its predecessor. It implements adapted prompt fine-tuning in the first stage, enhancing the model's ability to interpret and respond to user inputs. The second stage focuses on advanced prompt-label pair adaptation, leading to more precise and reliable segmentations.

The model excels in complex segmentation tasks, including boundary delineation and small structure identification. It also demonstrates superior generalisation across different medical imaging modalities and protocols, achieved through efficient training data utilisation and dynamic prompt sampling.

2.4 3D Data representation

Deep learning owes its popularity to the availability of large training datasets, the development of computational resources such as GPUs, and its success in extracting lower-dimensional representations of Euclidean data. This stands in various domains, including medical imaging. 3D data representations can be Euclidean, such as images, videos and text or Non-Euclidean, such as Point Clouds, Graphs and Meshes, Fig. 2.8. The latter can be of complex shape, represented by a variable size of unordered nodes or points and possibly a different number of node neighbours, making it more difficult to apply operations such as convolution on a graph than on an image [222], as we will see in the next section 2.5.

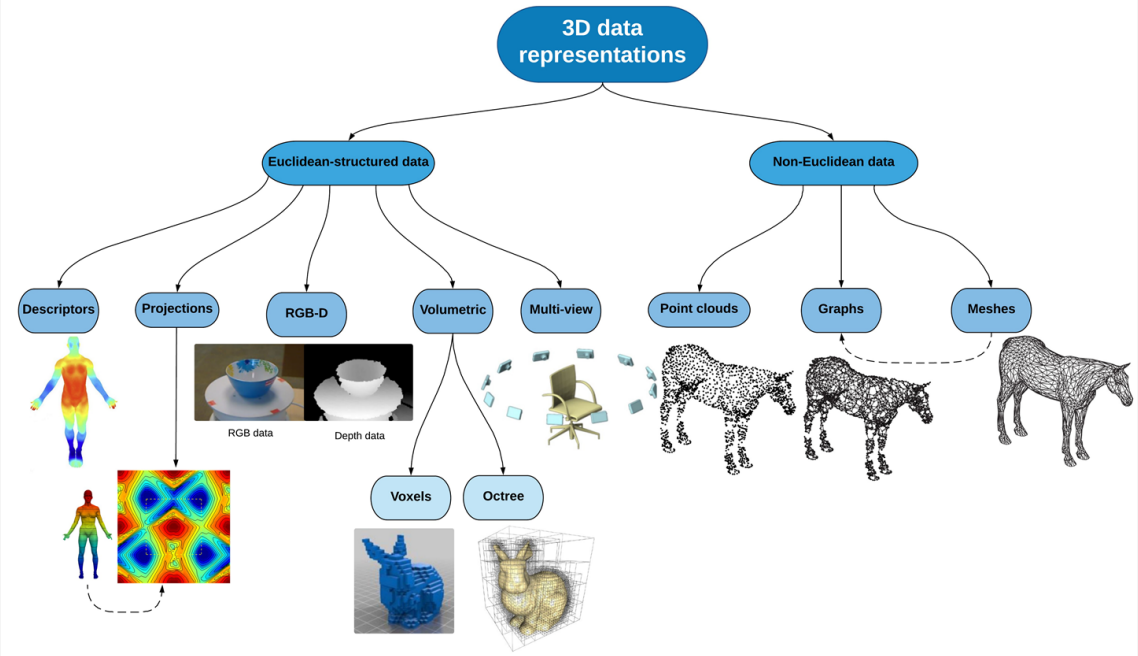


Figure 2.8: 3D Data Representations [2] Diagram shows an overview for the types of 3D data representation

2.4.1 Euclidean

Euclidean representations preserve their grid-structured data by having a common system of coordinates. In this category, the main 3D data representations are descriptors, projections, RGB-D data, volumetric data and multi-view data, Fig. 2.8.

Descriptors describe geometric or topological characteristics of the 3D shape including surface, texture, etc. or a combination [104] [237]. They are often referred to as a signature of the 3D shape that can be used to compare different 3D shapes. To better represent a 3D shape, descriptors can be combined with a learning-based model to

extract more hierarchical, distinctive features. Descriptors can be further categorised into local or global depending on what they describe, also based on whether the nature of the captured geometric information is intrinsic or extrinsic [2].

- Intrinsic descriptors are measured without leaving the surface, independent of external references. This could be measuring lengths and angles of vectors in the tangent space. The metric of a surface fully determines intrinsic quantities like geodesic distances. The Laplacian is, for example, an intrinsic operator. It is also worth mentioning that *Gaussian Curvature* is intrinsic; it compares the circumference of a geodesic circle to that of a circle in the plane with the same area and it is commonly known as the product of the two principal curvatures k1(x), k2(x) at a point x [217].
- Extrinsic descriptors treat a surface as an object embedded in the Euclidean space. The aforementioned principal curvatures are extrinsic as they are the eigenvalues of the rate of change of the surface normal (its derivative) [217].

3D descriptors can be divided into two categories: **signatures** and **histograms**, which are illustrated in Fig. 2.9. *Signatures* describe the 3D surface neighbourhood of a certain point, called support, using an invariant local Reference Frame. One or more geometric measurements are then measured and encoded for each point of the support. Although this method is highly descriptive, errors in the definition of the local Reference Frame, or perturbations, can modify the resulting descriptor. *Histograms* describe the support by creating histograms which encode local topological entities, such as mesh triangle areas or vertices, for instance, according to a certain quantity (normal angles, curvature, etc.). For example, the histogram could show the number of vertices with the same normal angle. Histograms are considered more robust compared to signatures but lack the descriptive power of signatures [176].

Projections The outcome of projecting 3D data into 2D space is a 2D grid with specific features, encapsulating key properties of the original 3D shape. Common methods are projecting 3D data into the spherical and cylindrical domains [25] [188]. Such projections are invariant to rotations around the principal axis of the projection. This makes 3D data processing easier due to the Euclidean grid structure of the resulting projections and takes advantage of the extensively researched learning models. However, for tasks involving dense correspondence, projections are not optimal due to the loss of information caused by the projection operation [191].

RGB-D images often come from RGB-D sensors, e.g., Microsoft's Kinect. It is a combination of an RGB image and its corresponding depth image. RGB-D data are simple, cheap and yet effective representations for 3D objects to be used for different tasks "such as recognition, pose regression, scene reconstruction, and correspondence"

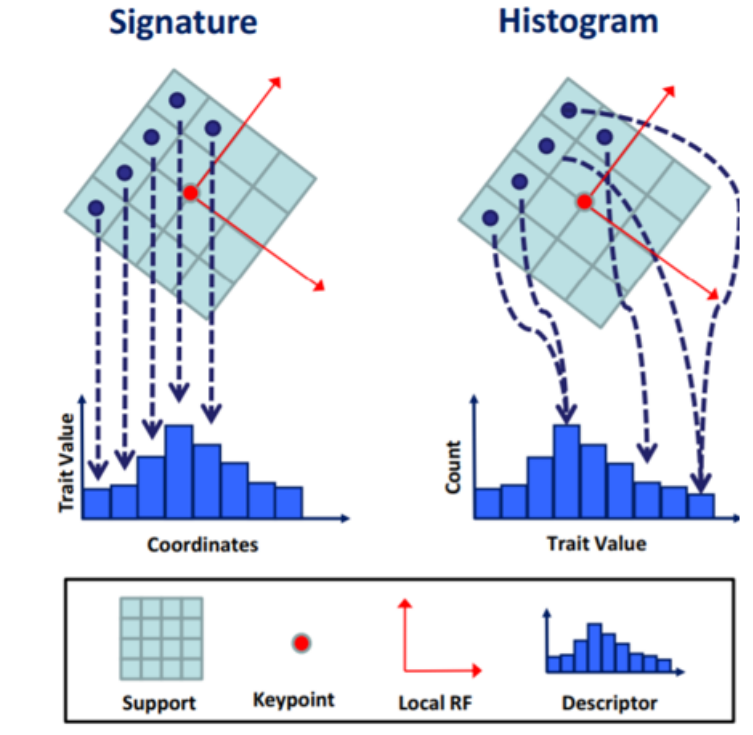


Figure 2.9: The two main categories of 3D descriptors: signatures and histograms [176].

[2].

Multi-view images are generated by projecting 3D objects to multiple 2D views. Su et al. [193] succeeded in using multi-view images and a novel pooling operation for 3D shape recognition. However, just like projections, it suffers from loss of information.

Volumetric; voxel grids and octrees The corresponding representation of pixels in 2D images are voxels in 3D. Voxels have a regular structure, which makes them suitable for CNNs and yields strong performance. Their main disadvantage is that they require high memory and calculation costs when the volumetric resolution is high. This limitation imposes a trade-off between memory consumption and quality because reconstruction quality critically depends on the grid resolution [76]. To address this, *Octrees* [198] are used by sparse voxel-based methods to represent the 3D object. Octree representations significantly reduce memory costs, but they still rely on the resolution of leaf nodes of octree to express fine details. Octrees are not easy to implement in learning-based approaches because their structure is inherently disconnected as the tree structure changes with occupancy.

2.4.2 Non-Euclidean

Unlike Euclidean data, there is no common system of coordinates, vector space structure, or shift-invariance. As a result, for tasks like recognition or correspondence, processing such data would usually be on a global scale, learning the features of the 3D object as a whole.

Point Clouds are easily obtained from 3D scanners. Notably, they are of irregular structure, making it hard to use with deep learning methods until Qi et al. introduced PointNet and PointNet++ for 3D classification and segmentation, using pooling operations that are order independent. "However, learning from irregular point clouds is still challenging and their method can only produce relatively coarse geometry.[76]"

Meshes are made up of points called vertices, pairs of vertices are connected via edges, and these make up small triangles called faces. To model a collection of 3D shapes, usually, one can be used as a template by which other shapes follow the same connectivity as that of the template. This would result in having a shape collection that is a deformation of a template shape.

> "Advantages of meshes over other shape representations should not be overlooked. Compared to voxels, meshes are more compact and better suited to representing finer surface details. Compared to points, meshes are more controllable and exhibit better visual quality - *Lin Gao, et al. 2019* [60]."

Graphs 3D meshes or point clouds can also be presented as graph-structured data where the nodes of the graph correspond to the vertices of the mesh or points. Graph-based networks generate edges that represent the connectivity between these vertices. Graphs can be directed or undirected [2]. A graph-based network looks like Fig. 2.10. Feature learning can be performed in a spatial or spectral domain, which will be further discussed in the coming section 2.5.

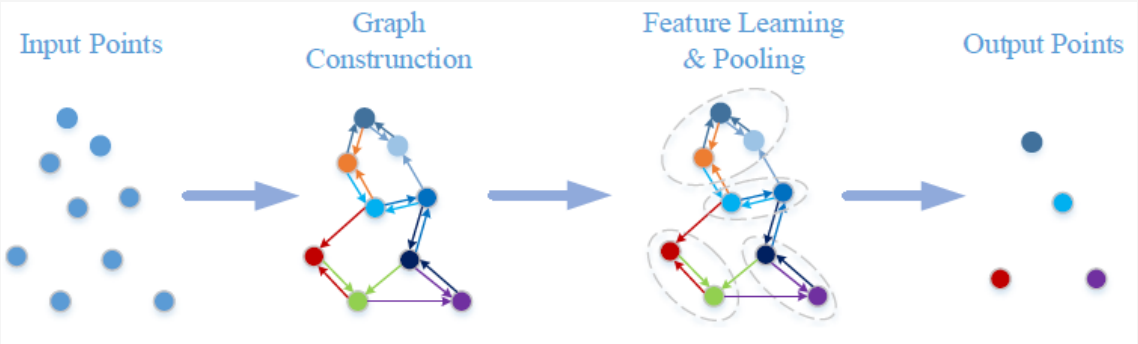


Figure 2.10: Illustration of a graph-based network [77]

2.5 Geometric Deep Learning

Geometric Deep Learning (GDL) is a term that encompasses deep learning methods designed to process data in non-Euclidean domains, specifically focusing on graphs and point clouds. GDL extends deep learning principles to handle irregularly structured data where the relationships between data points are not uniform or grid-like. Although seemingly new, this field has a rich history in deep learning development, particularly accelerated by the advances in CNNs.

The key challenge in GDL lies in adapting neural network operations, particularly convolutions, to these irregular structures. Unlike image data, where pixels are arranged in a uniform grid with fixed neighbourhoods, non-Euclidean data presents varying numbers of neighbours and irregular connectivity patterns. This fundamental difference has driven the development of specialised architectures and techniques.

The field has evolved rapidly since the first GNNs were proposed in 2005 [70]. Researchers have developed two main branches of solutions: methods for processing graph data and approaches for handling manifold data, particularly 3D point clouds.

2.5.1 Point Clouds

Point cloud processing methods are designed to handle the unique properties of 3D data, such as its unordered structure, irregular distribution, and varying densities. These methods share the common goal of extracting meaningful features for tasks like object detection, segmentation and classification. They can be broadly categorised into point Multi-Layer Perceptron (MLP)-based, convolutional, graph-based, and hierarchical approaches, emphasising different learning methods from spatial and geometric relationships in point clouds. Focusing on segmentation and classification tasks, we selected some of the significant models from each category to get an understanding of how these approaches work, but more comprehensive surveys can be found here [77, 117, 160, 209].

PointNet [155] is a pioneering MLP-based model that independently processes point clouds by learning features for each point, then aggregates these features using a symmetric max-pooling function to ensure permutation invariance. Its architecture comprises three key modules: a Spatial Transformer Network (STN) to identify important points, an RNN that treats the point cloud as a sequential signal and is trained to be invariant to the order of the points, and a max-pooling operation to aggregate the point features, maintaining permutation invariance. However, one major drawback of PointNet is its inability to capture local geometric structures, as it aggregates all point features globally, losing fine-grained shape details.

To address this limitation, PointNet++ [156] introduced a hierarchical approach by grouping points into local neighbourhoods and applying feature extraction at multiple scales. This design enables the model to capture local and global geometric features, significantly improving performance on complex shapes. However, the increased architectural complexity results in higher computational costs, especially when handling high-resolution point clouds.

Building on the idea of relational feature modelling, Dynamic Graph CNN (DGCNN) [218], the first graph-based method for point clouds, dynamically constructs local graphs in feature space. The key operation in DGCNN is EdgeConv, a graph convolution operation applied to each local neighbourhood, enabling the model to capture relationships between points dynamically at each layer. The k-nearest neighbour (k-NN) graph is recalculated during each iteration, allowing the model to learn evolving relationships between points throughout training adaptively. However, it also introduces computational overhead, slower training times, and higher memory requirements, particularly for dense point clouds.

PointCNN [120] was among the first to bring the convolutional paradigm to 3D point clouds without relying on voxelization or projecting onto 2D planes. This innovation opened the door for further development of convolutional methods such as KPConv [201], which is used for fine-grained segmentation, and FPConv [125] for classification. Additionally, MinkowskiNet [32] and SparseConvNet [71] focus on segmentation and classification, leveraging sparse volumetric representations to optimise memory and computation.

PointGCN [239], another graph-based model, operates by constructing graphs on the point cloud and applying graph convolutions to capture both local and global point relationships in a more efficient manner than earlier models like DGCNN.

More recently, PointNeXt [157] revisited the hierarchical principles of PointNet++ and modernised them with efficient training techniques, achieving state-of-the-art results in point cloud processing. Similarly, hierarchical methods like RandLA-Net [85] employed random sampling and attention mechanisms to efficiently process large-scale point clouds, offering a balance between computational efficiency and performance.

Emerging trends in this field include transformer-based architectures like Point Transformer [242], which uses self-attention to capture global and local dependencies and hybrid methods that combine point-based, graph-based, and grid-based representations for better performance and generalisation. However, challenges such as computational efficiency, handling noisy or incomplete data, and achieving robust generalisation across diverse datasets remain open areas for future research.

2.5.2 Meshes and Graphs

Mesh and graph-based processing methods are specifically designed to handle structured 3D data, such as triangular meshes or graph-like representations, which encode a shape's geometry and topology. These methods aim to extract meaningful features for segmentation, classification, and shape correspondence tasks. They build on the principles of GDL, leveraging the connectivity information inherent in meshes and graphs to learn representations that capture local and global geometric properties. Meshes can be easily represented as graphs, section 2.4.2. Thus, the models proposed for graphs can be used for meshes. Still, the opposite is not always possible because mesh-based methods often rely on geometric properties and assumptions (e.g., vertex positions, face normals) that do not exist in general graphs. This asymmetry makes graph-based models more versatile and applicable across different data domains. Therefore, we will talk about graph-based models in depth in the next section.

Meshes are particularly well-suited for tasks requiring explicit geometric details, such as shape analysis, segmentation, and deformation. MoNet [138] introduced a Gaussian mixture model to define local coordinate systems on non-Euclidean domains, enabling convolution-like operations on meshes for tasks such as segmentation and correspondence. Building on the idea of adapting traditional convolutions to irregular structures, PolyCNN [66] mapped mesh surfaces to planar domains using seamless toric covers, allowing the application of standard CNN operations to surfaces, which proved effective for classification and segmentation tasks.

MeshCNN [80] marked a significant advancement by introducing edge-based convolutional operators, focusing on learning edge features rather than vertex features. This approach improved mesh-based segmentation and classification by preserving mesh connectivity and geometric details. Similarly, methods like SpiralNet++ [69], and FeaStNet [208] took advantage of mesh structures by leveraging the ordering of local neighbourhoods for feature aggregation. These models are well-suited for learning geometric features from 3D shapes, enabling effective segmentation, correspondence and classification.

More recently, SubdivNet [86] leveraged mesh subdivision to create multi-resolution representations, enabling the learning of geometric features at varying levels of detail. This approach improved the model's capacity to handle meshes of varying complexities while maintaining computational efficiency. Emerging trends also include integrating self-attention mechanisms, as MeshTransformer [124] demonstrated, which captures both local and global dependencies across mesh structures, highlighting the growing interest in transformer-based architectures for mesh processing.

2.5.3 Graph Neural Networks

Graph Neural Networks (GNNs) extend the operation of Convolutional Neural Networks (CNNs) to non-Euclidean data, making them ideal for graphs and meshes. CNNs slide a kernel over a Euclidean grid (image) to extract local features, while GNNs extract information from a node's neighbourhood in a graph [222]. Unlike images, graphs have complex topologies, variable node orders, and numbers of neighbours, making convolution more challenging. GNNs address these challenges by employing permutation-invariant functions to aggregate information for graph-level tasks like classification. Message passing, the foundation of most GNN variants, involves (1) assigning a feature vector to each node, (2) aggregating neighbourhood information, and (3) updating node feature vectors [65]. This process is expressed mathematically as:

$$h_i^{(k+1)} = \sigma\left(h_i^{(k)}, \gamma\left(\{h_j^{(k)} \mid j \in N(i)\}\right)\right)$$

where h is a feature vector (embedding), i is the central node that aggregates information, j is the neighbouring node in layer k. σ and γ are the update and aggregate functions.

A general design pipeline for a GNN model is illustrated in Fig. 2.11.

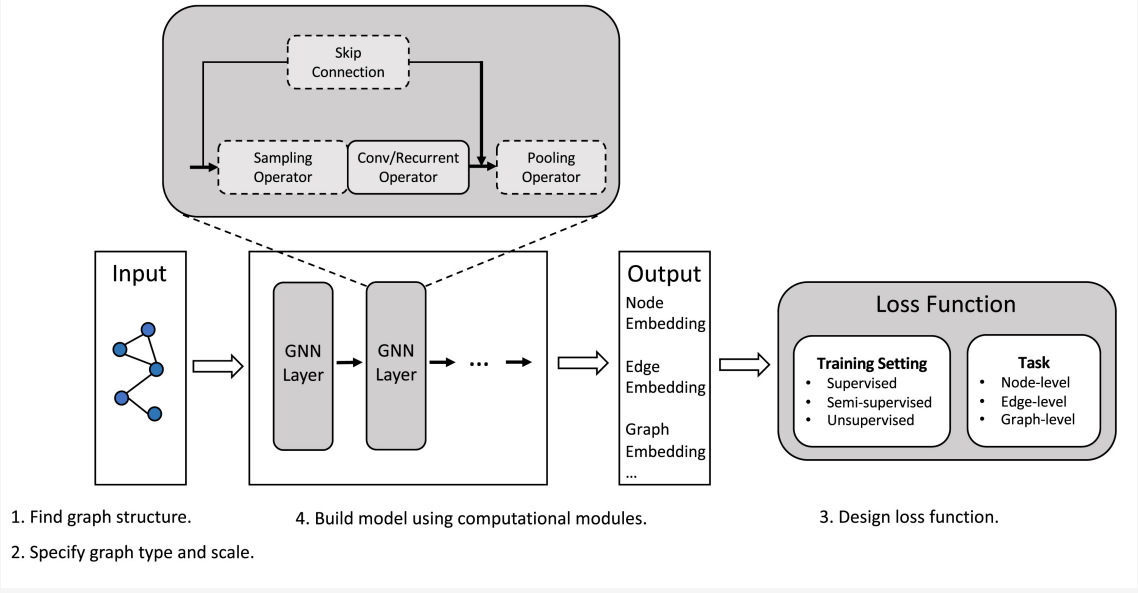
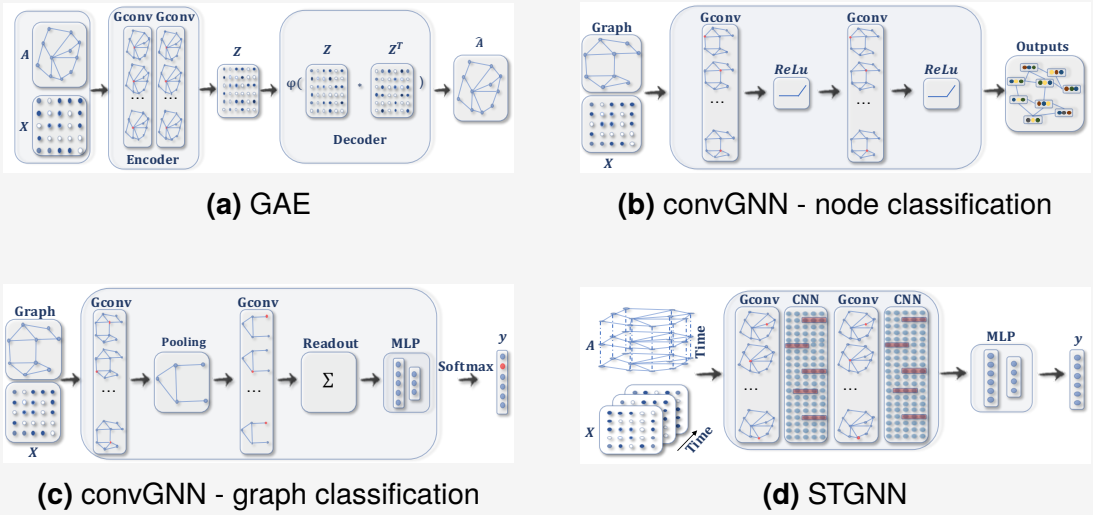


Figure 2.11: GNN model pipeline [245]

GNNs can be further categorised into four types: Convolutional Graph Neural Networks (ConvGNNs), Recurrent Graph Neural Networks (RecGNNs), Graph Autoencoders (GAEs) and Spatial-temporal Graph Neural Networks(STGNNs) [222], as shown

in Fig. 2.12. However, this section will focus only on ConvGNNs.



(a) GAE

(b) convGNN - node classification

(c) convGNN - graph classification

(d) STGNN

Figure 2.12: Types of GNNs [222]

It is important to revisit some graph theories that will help us understand GNN model components, specially the spectral filtering methods of ConvGNNs.

For simplicity, "spectral" means it can decompose a signal, image, audio or graph into a sum of wavelets or graphlets (simple elements). For instance, the Fourier Transform can represent any signal as a sum of sine and cosine waves of different frequencies. A common property is that decomposition produces simple elements that are orthogonal. However, with graphs, "spectral" implies eigen-decomposition of the graph Laplacian.

Graph Laplacian Matrix, has three forms [24]:

- **Combinational** $L = D - A$, where D is the degree matrix (a diagonal matrix) storing the degree of each vertex, meaning the number of edges attached to it and A is the adjacency matrix.
- **Symmetric Normalized** $L_sym = D^{1/2}LD^{1/2}$
- **Random Walk Normalized** $L_rw = D^{-1}L$

The graph Laplacian shows in what directions and how smoothly heat is transferred across a graph if we start with some "potential" in a specific node. No transfer occurs between nodes if an edge does not connect them, and the transfer is smooth when there is no sudden change in values among neighbours.

Eigen-Decomposition of L [24] Recall the *convolution theorem* that states that convolution in the spatial domain is multiplication in the frequency domain. Applying the Fourier Transform on graphs is not as trivial as with signals to transform them to the frequency domain. For signals, Discrete Fourier Transform (DFT) is applied to the signal by multiplying it by a DFT Matrix (a transformation matrix, basis). For graphs, eigen-decomposition is needed to obtain the basis. Therefore, Eigenvectors U (basis) of the graph Laplacian L and Λ the eigenvalues of L are obtained as follows:

$$L = U\Lambda U^T$$

The Eigen-decomposition of a graph Laplacian extracts subgraphs, clusters (communities) of nodes and mainly unsupervised learning tasks, whereas eigenvalues encode information regarding graph connectivity.

Below are the steps that need to be taken to design a GNN model, shown in Fig. 2.11:

1. **Set Graph Type** [245]:

Graphs can be **directed**, meaning the edge connecting a pair of nodes has a direction, or it can be undirected, resulting in an **undirected graph**.

Also, Nodes and edges can be of the same type, making a **homogeneous graph** or of different types, making a **heterogeneous graph**.

At last, if input features or graph structure change over time, the graph is **dynamic**.

2. **Add GNN Layer components:**

- **Sampling Module** GNN models aggregate messages for each node from its neighbourhood in the previous layer. The size of neighbours grows exponentially with the network's depth. To overcome this "neighbour explosion", sampling is needed. Another advantage of using a sampling module specially with large graphs, is to reduce storage and processing overhead of all neighbourhood information for each node when performing propagation. There are three kinds of graph sampling modules: node, layer, and subgraph sampling [245].

- **Convolution Operator** Existing ConvGNN methods will fall under spectral or spatial filtering methods. The main difference between the two categories is the filtering method employed and how locally processed information is combined.

  Spectral Filtering Methods define convolutions as spectral filtering, which is implemented as the multiplication of signals on the graph with eigenvectors of the graph Laplacian matrix.

  Spatial Filtering Methods define operations e.g. convolution and pooling in spatial domain. Usually, convolution is implemented by MLP over spatial neighbours, and pooling generates a coarsened graph by aggregating information from neighbouring points. Node features, such as colours, can be made of coordinates and intensities from a 3D scanner. Geometric attributes between a pair of points are usually set as the edge features.

  Spectral vs Spatial It is feasible to design a new graph signal filter, hence a new ConvGNN with a spectral model due to its theoretical foundation. However, spatial models are preferred over spectral models due to efficiency, generality, and flexibility issues; see Table 2.1.

- **Skip Connection** There are two orthogonal design views of a GNN: the number of layers or the network's depth and aggregation size. Increasing the depth (k layers) makes each node aggregate more information from neighbours that are k-hops away, achieving better results. However, there is a tradeoff, as this may degrade the GNN's performance due to propagating noisy information from an exponentially increasing number of neighbourhood nodes. On the other

Table 2.1: Spectral vs Spatial methods [222]

	Spectral	Spatial
Efficiency	less efficient, need to perform eigenvector computation or handle the whole graph at the same time	more scalable to large graphs as convolutions are performed in the graph domain via information propagation - performed in a batch of nodes
Generality	graph Fourier basis does not generalise well to new graphs. Any perturbations to a graph change its eigenbasis	graph convolutions performed locally per node, where weights can be shared across different locations
Flexibility	limited to undirected graphs	operates on edge inputs, directed graphs, signed graphs and heterogeneous graphs as they can be fed into the aggregation function

hand, designing a shallow network and increasing aggregation may cause over-smoothing because nodes would become more similar and require high computation and storage costs [249]. Therefore, adding skip connections is important to make GNN models deeper without compromising performance [24].

- **Pooling Module** A pooling layer usually follows a convolutional layer to extract more general information from nodes or get high-level subgraphs or graphs. However, graphs hold rich hierarchical structures that are important, specially for node-level and graph–level classification tasks. Therefore, it is common to design hierarchical pooling layers on graphs. Two kinds of pooling modules exist: direct and hierarchical [245].

3. **Design the loss function** [222]
   It is based on the type of task and the training framework.

   Task Type:

   - **Node-level** such as node classification, node regression and node clustering. Similar to section 2.2, node classification predicts a class/label for each node so that they are categorised. Node regression involves predicting a continuous value for a node. Node clustering aims to group similar nodes together, forming clusters. RecGNNs and ConvGNNs can extract high-level node representations using information propagation/graph convolution. End-to-end node-level tasks can be performed by GNNs, with a multi-perceptron or a softmax layer as the output layer.

- **Edge-level** Involve edge classification and link prediction, which require the model to classify edge types or predict whether an edge exists between two given nodes and possibly the strength of the connection.
- **Graph-level** includes graph classification, regression and clustering. GNNs are often combined with pooling and readout operations to obtain a compact representation on the graph level.

Training frameworks:

- **Semi-supervised learning** is often used for node-level and edge-level classification. The training input has only a few nodes labelled; the remaining majority is unlabelled. For testing, the model is expected to predict the labels for unlabelled and unseen new nodes.
- **Supervised learning** is often used for graph-level classification, and the training input is labelled data.
- **Unsupervised learning** is often used for graph embedding and node clustering. The model is expected to find patterns from unlabelled data.

2.6 Mesh Registration

Mesh registration is the process of aligning two or more meshes into a common coordinate system. It is a fundamental task in many applications, such as medical imaging, computer vision and 3D modelling. Registration methods can be broadly categorised into rigid and non-rigid methods, with some hybrid approaches combining elements of both.

Point cloud registration is more common and is therefore often used for mesh registration by which the template mesh topology is used for registered point clouds given correspondence has been established.

What does ”establishing correspondence across meshes” mean? For example, this step is often a prerequisite to applying PCA, as we will demonstrate in chapter 9. It means rearranging the order of the points in a point cloud or a mesh so that the first vertex from one subject's point cloud would be at the same anatomical location as the first vertex in any other point cloud from any other subject in the dataset. That way, when we average all point clouds, we get the mean shape, allowing us to perform PCA to visualise in 3D the top modes of variation.

Registration methods can be referred to as coarse (global) if an initial transformation is unavailable and fine (local) otherwise. Either method can follow a one-stage or a two-stage workflow. The one-stage approach attempts to predict the transformation between two cloud points in an end-to-end fashion without explicitly establishing the correspondence. Meanwhile, the two-stage approach involves determining correspondence before estimating the transformation. Four steps are commonly used for determining correspondence: (1)keypoints detection, (2)feature description, (3)descriptor matching (nearest neighbour) and (4)outlier detection.

Common notions:
- **Correspondence set K:** A set of point pairs (p, q) where p belongs to the source P and q to the target Q.
- **Transformation T:** A function applied to p in the form $T(p) = sRp + t$, where s is the scaling factor, R is the rotation matrix, and t is the translation vector.
- **Objective function $E(T)$:** Measures the alignment quality between transformed source points and target points.

2.6.1 Rigid Registration

Rigid registration entails that the registered object undergoes transformations limited to rotations and translations without any deformation. Therefore, it preserves distances between points after registration. A proposed taxonomy for rigid registration methods is shown in Fig. 2.13 from a recent review paper [132].

Umeyama's Algorithm [206] is a classic approach to solving the least squares problem for rigid registration. It computes the optimal translation, rotation, and scaling between two sets of corresponding points. The method minimises the following objective function over a correspondence set K:

$$E(T) = \sum_{(p,q)\in K} \|T(p) - q\|^2$$

This algorithm is effective when accurate correspondences are established.

ICP - Point to Point [150] aligns two point clouds P (source) and Q (target) iteratively:

1. **Correspondence finding:** Apply the current transformation T to P to align it with the target point cloud Q and associate each transformed point $T(p)$ with the nearest point in Q.
2. **Transformation update:** Minimize the objective function $E(T)$ defined over the correspondence set K to refine T:

$$E(T) = \sum_{(p,q)\in K} \|T(p) - q\|^2 = \sum_{(p,q)\in K} \|(Rp + t) - q\|^2$$

This process iteratively reduces the point-to-point distances between P and Q.

ICP - Point to Plane [30] extends the point-to-point ICP by leveraging surface normals for better alignment on structured surfaces. The objective function minimises the squared distances between transformed source points and planes defined by the normals of their corresponding target points:

$$E(T) = \sum_{(p,q)\in K} ((T(p) - q) \cdot n_q)^2$$

where n_q is the normal vector at q in the target cloud.

Fast Global Registration (FGR) with RANSAC [246] combines feature-based matching with RANSAC to estimate a robust initial transformation T:

1. **Feature Extraction:** Compute FPFH for points in P and Q, capturing local geometric properties.
2. **Feature Matching:** Establish correspondences by nearest-neighbour matching in the FPFH space, linking points in the source and target that share similar local geometric properties.
3. **Initial Transformation Estimation with RANSAC:**
   - **Random Subset Sampling:** Randomly select subsets of correspondences from P and Q to form a temporary sample set K_{sample} and compute candidate transformations T.
   - **Inlier Evaluation:** Apply T to all points in P and count inliers based on a distance threshold:
   $$T^* = \arg\max_T \sum_{(p,q)\in K} \mathbf{1}(\|T(p) - q\| < \text{threshold})$$
   where $\mathbf{1}(\cdot)$ is an indicator function that counts a point pair as an inlier if the distance between $T(p)$ and q is below the threshold.
   - **Maximising Inliers:** Repeat the sampling and inlier evaluation steps to find T^*, the transformation with the highest inlier count, best aligns P and Q.

2.6.2 Non-Rigid Registration

Non-rigid registration allows for deformations, making it suitable for cases where the source and target meshes undergo shape changes. Unlike rigid registration, it incorporates an elastic component to account for non-linear deformations, allowing transformations such as scaling, shearing, and bending. However, the added flexibility increases the degrees of freedom, making the optimisation process more challenging.

Coherent Point Drift (CPD) - Affine Registration [142] is a probabilistic point set registration method that models the source point cloud as a Gaussian Mixture Model (GMM), with the target points treated as observations drawn from this GMM. It moves the source points coherently, ensuring that nearby points move together smoothly while aligning to the target, like flowing in a consistent drift. This probabilistic framework enables CPD to handle both rigid and affine transformations by maximising the likelihood that the observed points correspond to the source point distribution.

- Objective Function: CPD minimises the Negative Log-Likelihood of the GMM, constrained by an affine transformation:

$$E(\mathbf{A}, \mathbf{t}) = \sum_{i=1}^{N} \log \sum_{j=1}^{M} \exp\left(-\frac{\|\mathbf{A} \cdot \mathbf{p}_i + \mathbf{t} - \mathbf{q}_j\|^2}{2\sigma^2}\right),$$

where $\mathbf{A}$ is an affine transformation matrix, and σ is a variance parameter.

- Expectation-Maximisation (EM) Optimisation:
  E-Step: Assigns probabilistic correspondences between source and target points, estimating which source points likely correspond to which target points.

  M-Step: Updates the transformation parameters A and t by maximising the maximum a posteriori (MAP) estimation and aligning the source with the target.

2.6.3 Deformable Registration

Permits local deformations while preserving spatial coherence It allows for complex, non-linear transformations, making it suitable for registering structures with significant shape differences.

CPD - Deformable Registration [142] extends affine CPD by allowing *non-rigid deformations* through spatial smoothness constraints. Points in the source can ”drift” to match the target while preserving local coherence using a regularised displacement field.

- Objective Function: The deformable CPD model includes an additional regularisation term to prevent unrealistic deformations, enforcing smooth transitions between neighbouring points:

  $$E(\mathbf{W}) = \sum_{i=1}^{N} \log \sum_{j=1}^{M} \exp\left(-\frac{\|\mathbf{p}_i + \mathbf{GW} - \mathbf{q}_j\|^2}{2\sigma^2}\right) + \lambda \mathrm{Tr}(\mathbf{W}^T \mathbf{GW}),$$

  where $\mathbf{W}$ is the displacement field, $\mathbf{G}$ is a Gaussian kernel defining spatial relationships between points, and λ is a regularisation parameter.

- EM Optimisation:
  E-Step: Probabilistically assigns correspondences between source and target points.
  M-Step: Updates W to minimise alignment error while enforcing smoothness, allowing the source to deform non-rigidly.

Thin Plate Splines (TPS) [18] is another popular method for deformable registration, particularly effective for smooth deformations and is computationally efficient for

moderate-sized datasets. TPS represents deformations as combining a global affine transformation and a localised warping function. It employs closed-form solutions for small datasets but often requires iterative solvers for larger problems.

The objective function minimises the bending energy of the deformation, which is modelled as:

$$E_{TPS} = \sum_{i=1}^{N} \|\mathbf{q}_i - (\mathbf{A} \cdot \mathbf{p}_i + \mathbf{t} + \mathbf{\Phi K})\|^2 + \lambda \text{Tr}(\mathbf{K}^T \mathbf{\Phi K}),$$

where:

- $\mathbf{A}$ is the global affine transformation.
- $\mathbf{K}$ is a kernel matrix computed based on the radial basis function.
- $\mathbf{\Phi}$ is the weight matrix for non-affine deformation components.
- λ controls the regularization to prevent overfitting.

2.6.4 Deep Learning Approaches

Deep learning-based point cloud registration methods leverage neural networks to automate alignment by learning transformation parameters between source and target point clouds. These methods can be grouped in different ways, as shown in Fig. 2.14 from a recent review paper [238].

Supervised methods rely on ground-truth transformations or correspondences during training, using loss functions that compare predicted and true transformations. For example, 3DRegNet [145] employs a compact architecture with convolutional and fully connected layers to efficiently predict rotation and translation. Innovations like FINet [224] and DetarNet [31] decouple rotation and translation estimation, with FINet processing them in parallel and DetarNet solving them sequentially. Despite their accuracy, these methods heavily depend on extensive labelled datasets, making them less practical in real-world applications.

Unsupervised methods avoid the need for ground-truth labels by minimising discrepancies between features or point clouds. Correspondence-free approaches, such as PPF-FoldNet [45] and UGMM [88], focus on global alignment by using loss functions like the Chamfer Distance to measure bidirectional discrepancies between point clouds or Gaussian mixtures. Correspondence-based methods, such as CEMNet [98] and RIENet [187], refine alignment by estimating reliable correspondences and introducing specialised losses, including neighbourhood consensus and spatial consistency losses.

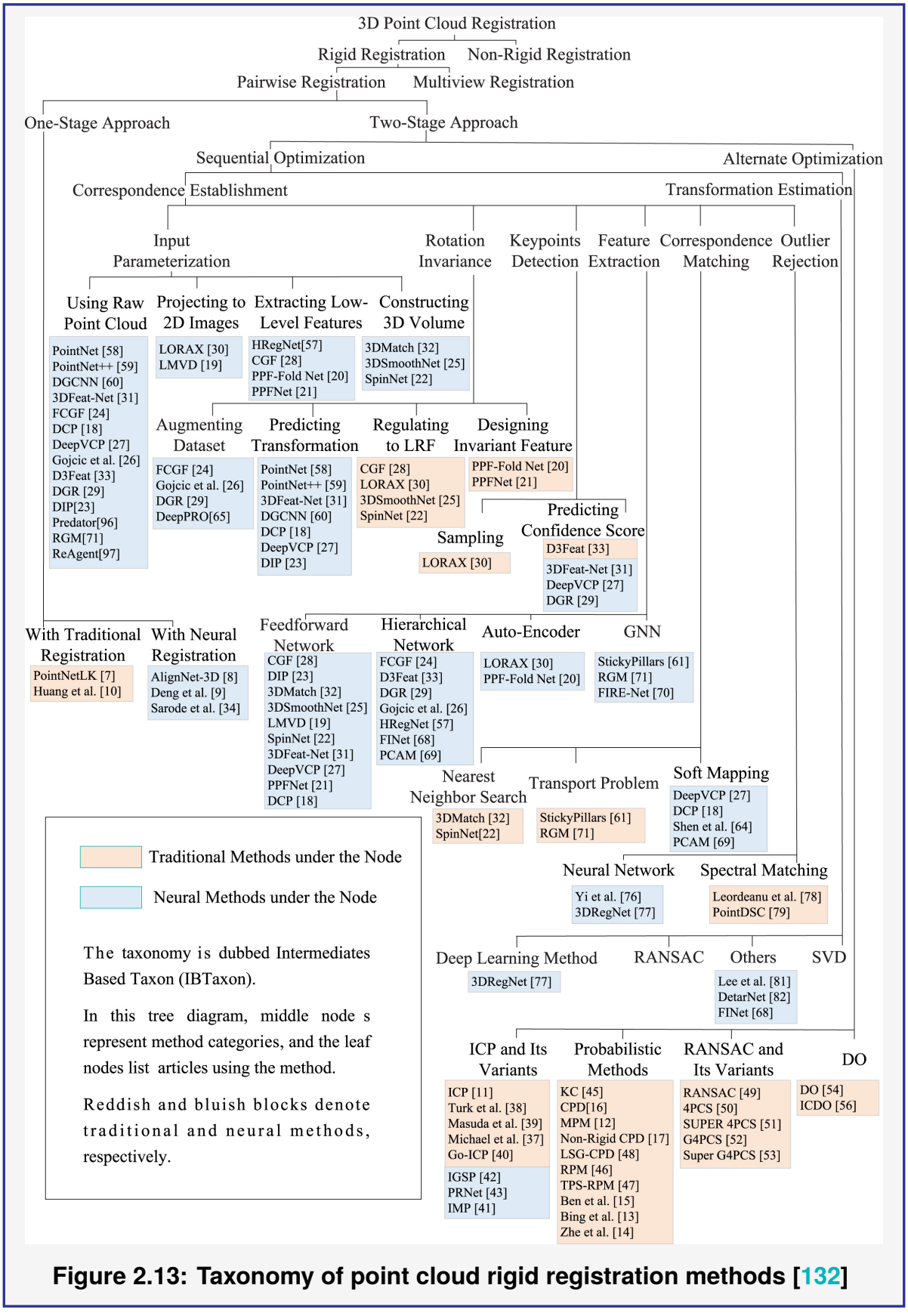


Figure 2.13: Taxonomy of point cloud rigid registration methods [132]

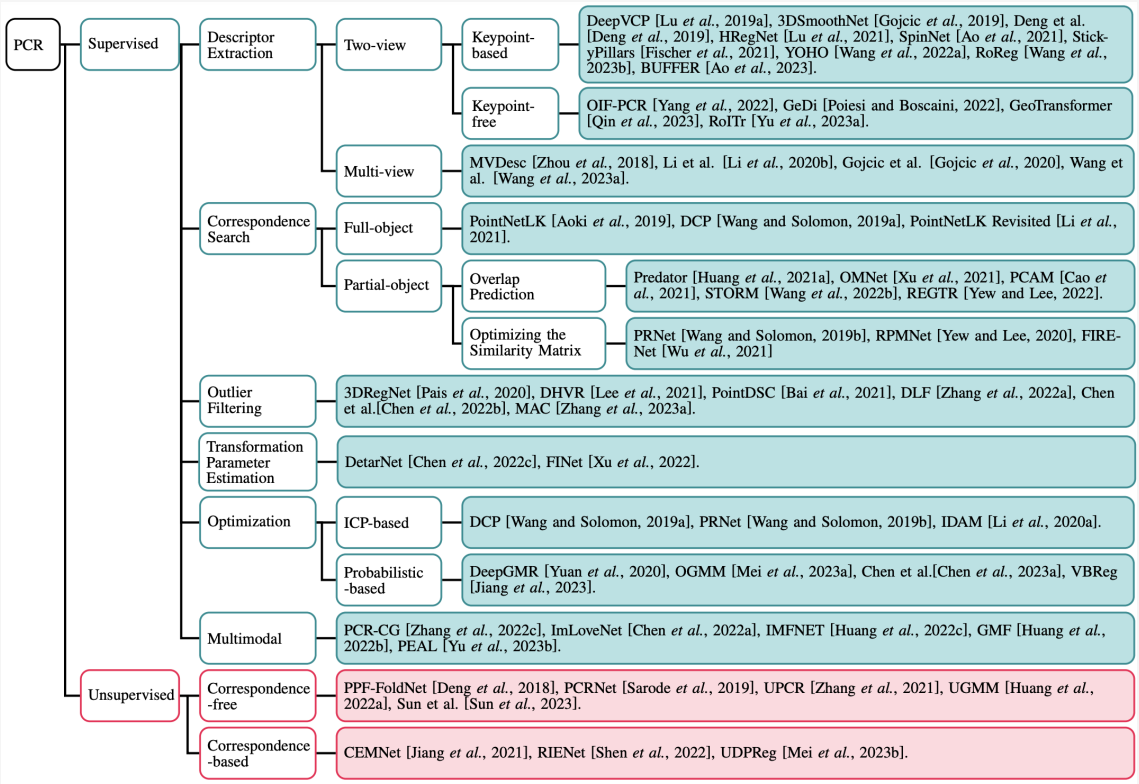


Figure 2.14: Taxonomy of deep learning based Point Cloud Registration methods, denoted as PCR in the figure [238]

2.7 Evaluation metrics

Evaluation metrics that can be used to assess the methods used for tasks described in section 2.2 and for mesh or point cloud registration methods from section 2.6 are presented below.

Evaluating segmentation quality: Given A, the predicted segmentation set (e.g., the set of voxels or points labelled as part of a specific class by the model) and B, the ground truth segmentation set (e.g., the set of voxels or points belonging to the same class in the ground truth annotations).

- **Mean Accuracy:** Accuracy is computed as the ratio of correctly classified pixels for that class to the total pixels of the same class in the ground truth. Then, the average of per-class accuracies is calculated as:

$$\text{Mean Accuracy} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{True Positives}_c}{\text{True Positives}_c + \text{False Negatives}_c}$$

  where C is the total number of classes (or labels).True Positives$_c$ is the number of correctly predicted pixels for class c and False Negatives$_c$ is the number of pixels belonging to class c in the ground truth but missed in the predictions.

- **Mean Intersection over Union (mIoU):** measures the overlap between the predicted and ground truth regions for a given class. First, IoU is calculated for each class or label, and then the average is computed across all classes:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives} + \text{False Negatives}}$$

- **Precision:** Measures the proportion of correctly predicted positives out of all predicted positives:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \textbf{False Positives}}$$

- **Recall:** Measures the proportion of actual positives correctly identified:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \textbf{False Negatives}}$$

- **Dice score:** Also known as the F1-score in segmentation contexts, the dice score measures the overlap or similarity between two segmentations (e.g., predicted and

ground truth regions):

$$\text{Dice score} = \frac{2|A \cap B|}{|A| + |B|}$$

The Dice coefficient is symmetric, meaning the order of A and B does not matter. This implies that the dice score will be the same if an object in the image is "under" or "over"-segmented by the same amount.

- **Mean Average Precision (mAP):** Computed as the area under the precision-recall curve for each class, then averaged over all classes.
- **Volumetric and Surface-Based Metrics:** These metrics measure similarity and distances between segmented volumes or surfaces, reported per class label. Therefore, these are also used to assess mesh registration by replacing A and B by the sets of deformed ($\mathcal{V}_{\text{deformed}}$) and target ($\mathcal{V}_{\text{target}}$) vertices.
  - **Dice score and IoU:** Same as above, dice measures the similarity between two meshes.
  - **Mean Surface Distance (MSD):** first, the surface distance SD_a, B between point a on surface A and surface B is calculated as the minimum of the Euclidian norm for all the points b on surface B. This finds the closest point on B to the point a:

    $$SD_{a,B} = \min_{b \in B} \|a - b\|_2$$

    Then, the total surface distance SD_{AB} between A and B is calculated as the sum of the surface distances for all points in A. Note that the distance is not assumed to be symmetric, therefore, the surface distance SD_{BA} needs to be calculated from B to A:

    $$SD_{AB} = \sum_{a \in A} SD_{a,B}$$

    Finally, the Mean Surface Distance, which is the average closest-point distance between surfaces A and B is obtained by taking the mean over all N_A points in A:

    $$MSD_{A,B} = \frac{1}{N_A} SD_{AB}$$

  - **Root Mean Squared Error (RMSE):** A distance metric that emphasises larger errors and quantifies the average deviation between corresponding points as

    $$RMSE_{A,B} = \sqrt{\frac{1}{N_A} \sum_{a \in A} \left(\min_{b \in B} \|a - b\|_2 \right)^2} = \sqrt{\frac{1}{N_A} \sum_{a \in A} {SD_{a,B}}^2}$$

  - **Hausdorff Distance (HD):** represents the maximal surface distance calculated

between the surfaces A and B. This quantity is not necessarily symmetric.

$$HD_{AB} = \max_{a \in A} SD_{a,B}$$

Assessing shape classification model performance: Mean accuracy, precision and recall are commonly used, defined the same as above and ROC curves as in Fig. 2.15.

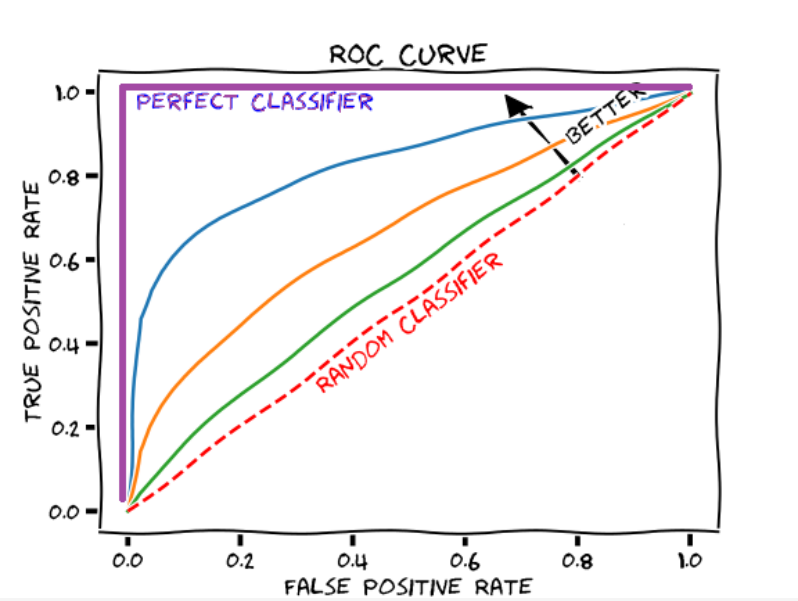


Figure 2.15: AUROC [159]

Assessing shape regression model performance: Regression models are commonly evaluated using metrics that quantify their predictive accuracy.

- **Mean Absolute Error (MAE)** measures the average magnitude of errors in predictions, given by

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|,$$

  where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the number of samples.

- **coefficient of determination (R^2)**, which assesses the proportion of variance explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2},$$

  where $\bar{y}$ is the mean of the actual values. Higher R^2 values and lower MAE indicate better model performance.

Chapter 3

Datasets

This chapter provides an overview of the image acquisition and pre-processing methods used to create the datasets for the studies in the succeeding chapters. Section 3.1.1 to Section 3.1.5 detail the acquisition protocols and metadata available for each dataset: UKBB, CamCAN, IXI, OASIS3 and E-HVols, spanning a variety of MRI scanners and protocols across healthy and diseased cohorts. Section 3.2 describes the pre-processing pipeline applied to align these datasets, including skull stripping, bias field correction, and segmentation. However, a custom pipeline was developed for the E-HVols dataset presented separately in chapters 8 and 9. Lastly, Section 3.3 introduces the graph representation of segmented structures, detailing the node features and edge attributes used to analyse shape information, which is used for all datasets, including the E-HVols.

3.1 Image acquisition and meta-data

3.1.1 UK Biobank (UKBB)

T1-weighted images of healthy volunteers were collected at the UKBB imaging centre using a 3T Siemens Skyra scanner with a 32-channel receiver head coil. Images were acquired following the 3D MPRAGE protocol; R=2, TR=2000ms, TE=385ms, TI=880ms; FOV=208x256x256mm; 1mm isotropic; Duration 4minutes 54s. This dataset was used for training our models because it is the largest of all our datasets. It consists of **13,749** samples, 47 % females, and the mean age of the volunteers is 61 years, ranging from 44 to 73 years old.

3.1.2 Cambridge Centre for Ageing and Neuroscience (Cam-CAN)

All subjects were scanned at the Medical Research Council Cognition and Brain Sciences Unit in Cambridge, UK, using a 3T Siemens TIM Trio scanner equipped with a 32-channel receive head coil. T1-weighted images used were acquired following the 3D MPRAGE protocol; TR=2250ms, TE=2.99ms, TI=900ms; FA=9 deg; FOV=256x240x192mm; 1mm isotropic; GRAPPA=2; TA=4mins 32s. Cam-CAN has **652** samples, of which 51% are females, with a mean age of 54, ranging from 18 to 88 years old.

3.1.3 Information eXtraction from Images (IXI)

All individuals participating were scanned at the Guy's Hospital using a 1.5T Philips scanner, Hammersmith Hospital, which used a 3T Philips system, and the Institute of Psychiatry, which used a GE 1.5T system. The following were the acquisition parameters: TE = 4.603ms, TR = 9.813ms, spatial resolution = 0.94 × 0.94 × 1.2mm3 in T1-weighted images. IXI has **563** samples, of which 58% are females, with a mean age of 49, ranging from 20 to 86 years old.

3.1.4 Open Access Series of Imaging Studies (OASIS-3)

We also use the OASIS-3 dataset with 716 cognitively normal participants and 318 participants who reached various stages of cognitive decline during the study for Alzheimer's disease (AD) classification [112]. The cognitive status is reflected in the clinical dementia rating (CDR) that accompanies the imaging dataset, with subjects receiving a score of 0 for healthy, 0.5 for very mild dementia, 1 for mild dementia, 2 for moderate dementia and 3 for severe dementia [140]. The CDR is collected in clinical sessions, separate from the imaging sessions; meaning sessions must be 'matched' to get an {image, CDR score} pair. We match the clinical diagnosis closest in time to each scan

before filtering out samples where the absolute time difference between the scan and clinical assessment is greater than 365 days. We only use one scan per subject to avoid difficulties in assigning scans to training, validation, and testing, leaving 1,084 unique scans. We excluded 50 samples because their sex or age information was missing. The final set of **1,034** comprises 716, 188, 111, 18 and 1 samples, for CDR of 0, 0.5, 1, 2 and 3, respectively. We binarise CDR to 0 and 1 (for CDR scores 0.5, 1, 2 and 3). The average age within the final set is 72, ranging from 42 to 97. Females constitute 55% of the final set.

3.1.5 Egyptian Healthy Volunteers (E-HVols)

Data was acquired from a cohort size of **744** subjects that have standard aortic dimensions and shape. Hence, these were used for training and evaluating aortic segmentation models in chapter 8 and for investigating different methods of mesh registration in chapter 9. The data collected from the 744 subjects come from two groups: 625 who volunteered for the study, and the remaining had complaints that did not affect the aortic shape. Therefore, for chapter 9, where we explore the data and run baseline prediction models, we excluded that latter group to ensure we only study healthy subjects. Out of the volunteered 625, 26 were labelled as unhealthy by the clinicians at the AHC.

Informed consent was obtained from the self-admitted volunteers before conducting the study. Under the Egyptian Healthy Volunteers Study E-HVol [1] – approved by the AHC research ethics committee `(20160401MYFAHC_HVOL)`, cardiovascular imaging (CMR, ECHO and ECG), next-generation sequencing (NGS) and other clinical assessments were acquired, at the AHC. CMR imaging scans were performed on E-HVols using a Siemens Magnetom Aera 1.5 Tesla MRI scanner at AHC, acquired using the `tfi3D-fs-free` breathing navigator sequence. The pixel spacing and slice thickness of the images were 2.0mm and 1.6mm, respectively.

3.2 Image post-processing

The UKBB data comes pre-processed with already extracted meshes for 15 subcortical brain structures[1], see section 2.1.2. We apply our own processing pipeline to Cam-CAN, IXI and OASIS-3 to match UKBB as closely as possible:

1) Skull stripping with ROBEX v1.2[2] [91]

2) Bias field correction with N4ITK[3] [204]

3) Sub-cortical brain structure segmentation and meshing using FSL FIRST [4] [149].

For the E-HVols dataset, we develop a custom pipeline for segmentation and meshing described in chapter 8 and chapter 9.

[1] Brain stem, left/right thalamus, caudate, putamen, pallidum, hippocampus, amygdala, accumbens-area

[2] https://www.nitrc.org/projects/robex

[3] https://itk.org

[4] https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FIRST

3.3 Graph representation

The extracted shapes from segmenting the images are represented by meshes as undirected graphs composed of nodes, connected by edges forming triangular faces.

Node representation The meshes are defined by a set of nodes and edges, where both can carry additional information. Nodes can encode arbitrary feature vectors, from spatial information such as mesh coordinates to more complex, geometric feature descriptors. In computer vision, hand-crafted features based on carefully designed descriptors have been largely abandoned in the end-to-end deep learning paradigm [12]. However, in the case of shape analysis, we believe there is value in sophisticated, geometrical feature extractors, especially when there is limited training data. We evaluate the use of Fast Point Feature Histograms (FPFH) [174] as node features and compare these with positional node features in the form of Cartesian coordinates and without node features (realised by setting constant values).

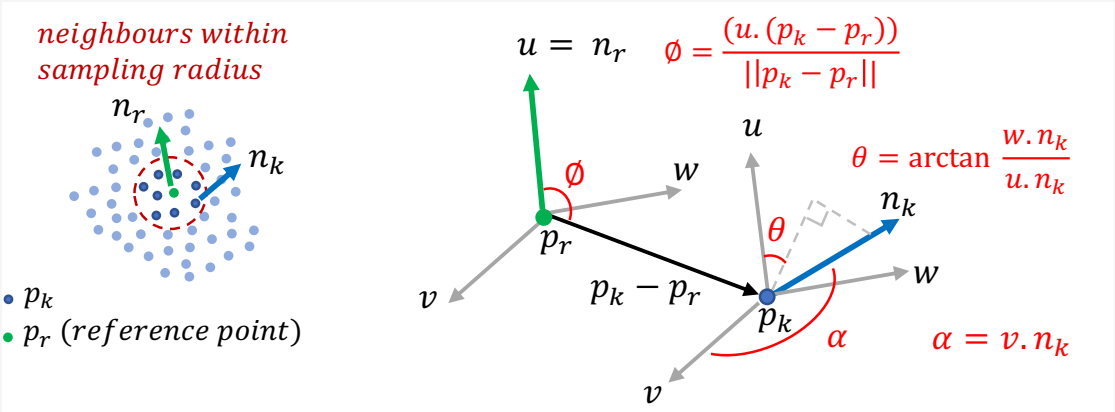


Figure 3.1: PFH calculation k-neighbourhood (left), Darboux frame definition and obtaining angular features (right).

To calculate FPFH features on a 3D point cloud or mesh, the following steps are performed:
First, a Point Feature Histogram (PFH) is computed for each query point p_r; Fig. 3.1:

1. Select all neighbouring points p_k inside a 3D sphere of radius r centred at p_r (k-neighbourhood)
2. For each point pair (p_r, p_k) in the k-neighbourhood, estimate their normals n_r and n_k.
3. For each pair, define a Darboux frame as:

$$u = n_r$$

$$v = (p_k - p_r) \times u$$

$$w = u \times v$$

4. Compute three angular features:
   Angle between normal n_k and v axis $\alpha = v \cdot n_k$
   Angle between normal n_r and point difference vector $\phi = \frac{u \cdot (p_k - p_r)}{\|p_k - p_r\|}$
   Angle between normal projections $\theta = \text{atan2}(w \cdot n_k, u \cdot n_k)$

Next, a Simplified Point Feature Histogram (SPFH) is computed for each point by binning the (α, ϕ, θ) features of its neighbours [175].

Finally, the FPFH feature at point p_r combines both its own SPFH and the weighted SPFH values of its neighbours. This weighting scheme considers the spatial distances between points, giving less influence to more distant neighbours, Fig. 3.2. The FPFH for point p_r is calculated as:

$$FPFH(p_r) = SPFH(p_r) + \frac{1}{N} \sum_{k=1}^{N} \frac{SPFH(p_k)}{\|p_k - p_r\|}$$

where:
- N is the number of neighbours within the sampling radius r from p_r.
- $\|p_k - p_r\|$ is the Euclidean distance between points p_k and p_r. The weight $\frac{1}{\|p_k - p_r\|}$ ensures that closer neighbours have more influence on the final histogram.

In our implementation, we used a sampling radius of 10mm and a maximum of 100 neighbours (i.e. k=100), which balances feature descriptiveness and computational efficiency well (unless mentioned otherwise, in the following chapters).

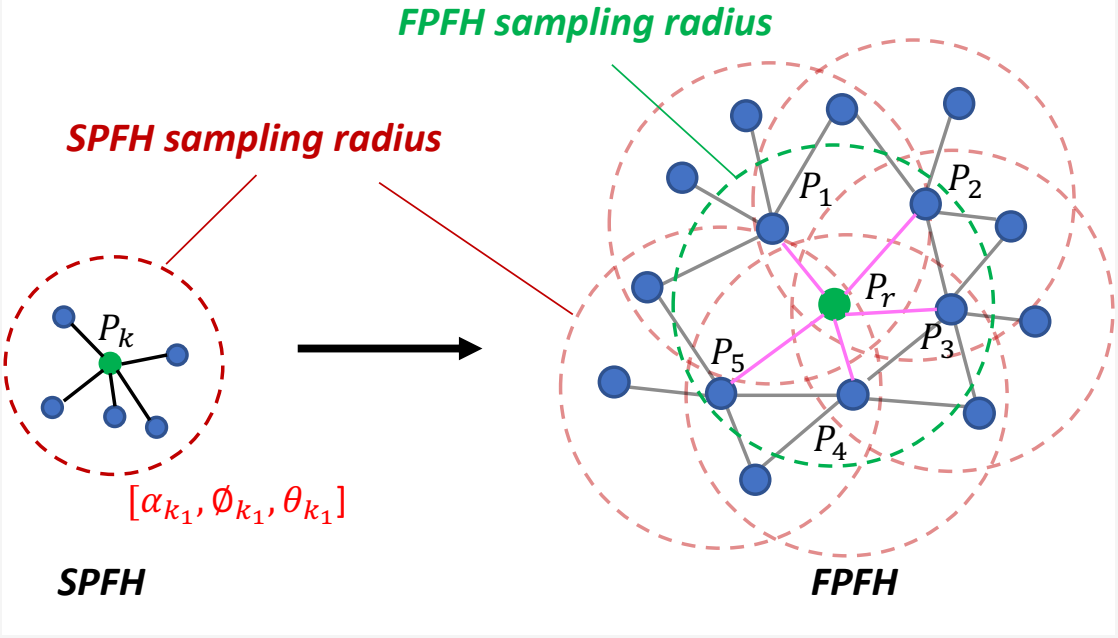


Figure 3.2: FPFH representation using nearby SPFH

Edge representation In addition to the node features, we encode edge attributes in terms of relative spherical coordinates between two nodes. Edge attributes are processed only within SplineCNN layers but otherwise ignored in both GCNConv and GraphConv layers (which we will use in our comparative study in the next chapter 4), as these can only use edge weights and not attributes.

Chapter 4

A Comparative Study of Graph Neural Networks for Shape Classification in Neuroimaging

Graph Neural Networks (GNNs) have emerged as a promising approach for analysing non-Euclidean data such as meshes. In medical imaging, mesh-like data plays an important role in modelling anatomical structures, and shape classification can be used in computer-aided diagnosis and disease detection. However, with a plethora of options, the best architectural choices for medical shape analysis using GNNs remain unclear.

We conduct a comparative analysis to provide practitioners with an overview of the current state-of-the-art in geometric deep learning for shape classification in neuroimaging. Using biological sex classification as a proof-of-concept task, we compare the performance of three alternative convolutional layers and reinforce the importance of data augmentation for graph-based learning. We find that using Fast Point Feature Histograms (FPFH) as node features substantially improves GNN performance and generalisation to out-of-distribution data. We then confirm these results hold for a clinically relevant task, using the classification of Alzheimer's disease.

This chapter is based on the following publication:

> *Shehata, N., Bain, W., and Glocker, B. "A Comparative Study of Graph Neural Networks for Shape Classification in Neuroimaging". In: Proceedings of the First International Workshop on Geometric Deep Learning in Medical Image Analysis. Vol. 194. Proceedings of Machine Learning Research. PMLR, 18 Nov 2022, pp. 160–171*

4.1 Introduction

Geometric deep learning generalises classical neural network models to non-Euclidean domains such as point clouds, graphs, or meshes [222]. It has therefore become popular across various fields ranging from computer vision [245] and physics [189], to healthcare topics [42] such as disease prediction [103], drug discovery [118], and brain connectome analysis [107].

A recent study [177] investigated the expressiveness of mesh representations for disease classification. We complement these findings by conducting a comparative study evaluating different GNNs to classify anatomical meshes extracted from neuroimaging data. We propose a simple yet effective multi-graph architecture with a shared submodel for learning shape embeddings (see Fig. 4.1). We compare different graph convolutional layers are compared; GCNConv [109], GraphConv [139], and SplineCNN [57]. In all cases, we observe substantial performance improvements when using FPFH as node features, which, to our knowledge, has not been explored before. We also investigate the effect of data augmentation, finding improvements in generalisation to data from new domains. Our findings on the proof of concept task of biological sex classification are confirmed on the clinically relevant diagnostic task of Alzheimer's disease classification.

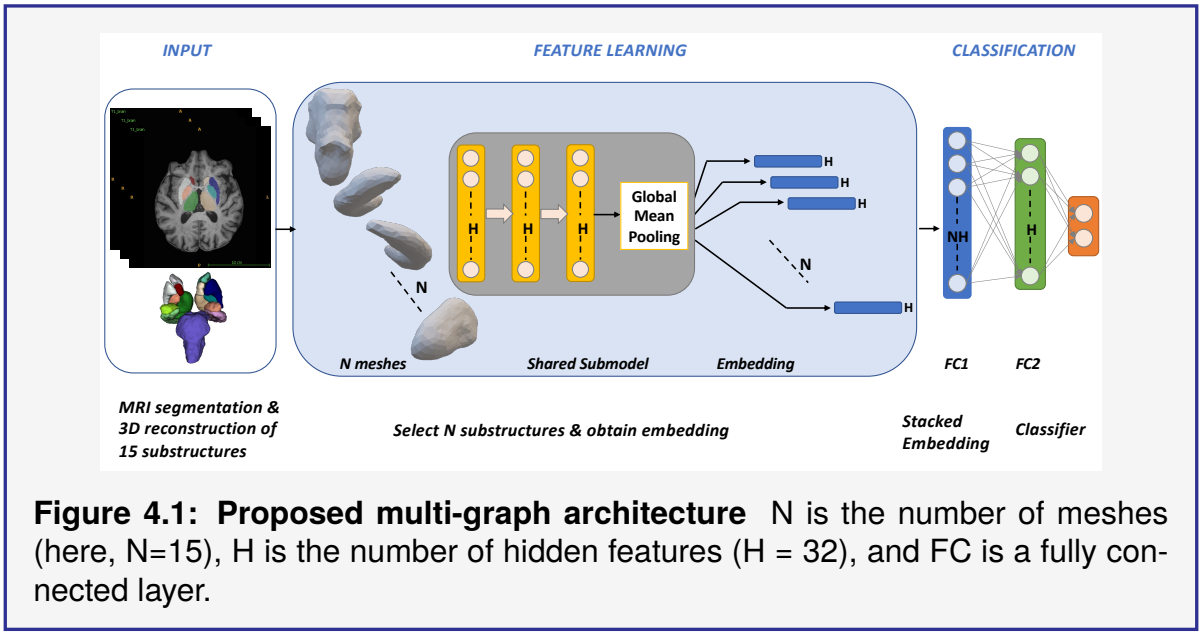


Figure 4.1: Proposed multi-graph architecture N is the number of meshes (here, N=15), H is the number of hidden features (H = 32), and FC is a fully connected layer.

4.2 Mutli-graph Architecture

As the field of geometric deep learning has expanded, the architectural choices available to practitioners have proliferated. Here, we outline our approach in three key aspects: the type of convolutional layer used, the number of convolutional submodels, and the type of geometric features encoded at the node level.

4.2.1 Selected graph convolutional operators

Graph convolutional operations are analogous to Convolutional Neural Networks (CNN) operations on images, respecting the additional invariants that arise in this domain, permutation invariance being key due to the artificial ordering of nodes that arises when representing graphs. As shown in [20], many GNNs follow a blueprint of 'message passing' [65], whereby node features are updated using an aggregation on the features of nodes in their neighbourhood, but there is significant variance in how this is done. In this paper, we compare three seminal graph convolutional layers from the literature: GCNConv [109], GraphConv [139], and SplineCNN [57]. We refer the readers to the original papers for details. Existing literature has compared GCNConv and GraphConv [139, 225], and we extend this to medical imaging.

GCNConv [109] proposed that the update and aggregate functions are combined into one operation; the update is incorporated by adding a self-loop for a particular node and including it in the summation, and the aggregate function is the sum operation of neighbour embedding normalised by node degrees where a weight matrix first transforms node features. This can be described mathematically as:

$$h_i^{(k+1)} = \sigma\left(\sum_{j \in N(i) \cup i} \frac{1}{\sqrt{\hat{d}_i}\sqrt{\hat{d}_j}} h_j^{(k)} W^{(k)}\right)$$

Where:
- $h_i^{(k+1)}$ is the updated feature vector of node i at layer $k+1$
- σ is a non-linear activation function
- $N(i)$ is the set of neighbours of node i
- $\hat{d}_i = d_i + 1$ is the degree of node i plus one (to account for the self-loop)
- $h_j^{(k)}$ is the feature vector of node j at layer k
- $W^{(k)}$ is the learnable weight matrix at layer k

GraphConv The GraphConv layer proposed by [139] omits the normalisation step of the GCNConv, as applying neighbourhood normalisation can reduce the ability of GNNs to distinguish certain graph structures [139, 225]. Instead, it introduces a simple skip-connection to increase the self-weight of each node.

$$h_i^{(k+1)} = \sigma\left(\sum_{j \in N(i)} h_j^{(k)} W^{(k)} + h_i^{(k)}\right)$$

Where:
- $h_i^{(k+1)}$ is the updated feature vector of node i at layer $k+1$
- σ is a non-linear activation function
- $N(i)$ is the set of neighbours of node i
- $h_j^{(k)}$ is the feature vector of node j at layer k
- $W^{(k)}$ is the learnable weight matrix at layer k
- $h_i^{(k)}$ represents the skip-connection

SplineCNN SplineCNN proposed by [57] aggregates features h_j in the spatial domain, leveraging the local support properties of B-spline basis functions, where for all inputs outside a known interval they evaluate to zero [152]. The continuous kernel function g_l, using B-spline bases, is parametrised by a fixed number of trainable weights $w_{p,l}$, where l indexes input feature maps M_{in} and therefore the computation time is independent of kernel size. This leads to speed improvements over comparable methods. Another key difference from other spectral methods is that information is not only encoded in connectivity, edge weights and node features but also in the relative position of nodes. Therefore, SplineCNN can incorporate edge features in feature aggregation by which the kernel function g maps pseudo coordinates $u(i,j)$ (e.g. cartesian, spherical or polar node coordinates) to a scalar that is used as a weight for feature aggregation [57]. The spatial convolution operator for a central node i is given by:

$$(h * g)(i) = \frac{1}{|N(i)|} \sum_{l=1}^{M_{in}} \sum_{j \in N(i)} h_l(j) \cdot g_l(u(i,j))$$

This can be reduced to:

$$(h_l * g_l)(i) = \sum_{\substack{j \in N(i), \\ p \in P(u(i,j))}} h_l(j) \cdot w_{p,l} \cdot B_p(u(i,j))$$

Where:
- $h_l(j)$ is the l-th feature of node j
- g_l is the continuous kernel function for the l-th input feature map
- $u(i,j)$ represents the pseudo coordinates between nodes i and j
- $N(i)$ is the set of neighbours of node i
- M_{in} is the number of input feature maps
- $w_{p,l}$ are trainable weights
- B_p is the product of B-spline basis functions
- $P(u(i,j))$ is the set of non-zero basis functions for $u(i,j)$

4.2.2 Shared versus non-shared submodel

Multiple subcortical structure subgraphs may be extracted simultaneously from a single sample brain scan, but one must also choose how to utilise them. One option is to combine them into a single multigraph per sample [27, 214]. However, it might not be obvious how to define edges between graphs of different anatomical structures.

Alternatively, as in this study, each subgraph can either be input to a single, shared GNN or multiple GNNs, with the results then combined into a sample-level output. The approach using the single **shared GNN** learns from all 15 subgraphs. Here, the weights of the graph convolutional layers are shared across brain structures. The shared submodel approach is parameter-efficient and aims to learn generic shape features. While the other, the **non-shared GNN** (also referred to as *non-shared submodel*), is inputting each subgraph to a separate GNN, i.e. the number of (sub) GNNs is equal to the number of subgraphs per sample [84]. The latter approach allows each (sub) GNN to learn structure-specific embeddings, whilst the former encourages the GNN to generalise the learning across structures.

Initially, we tested both a single shared and structure-specific GNN submodel, finding that the performance was comparable. Using a shared submodel significantly reduces the number of parameters. Given considerations on neural networks training time [116], cost [219], and environmental impact [192], our preliminary results led us to use a shared GNN in this study: each brain substructure is passed to the shared submodel to obtain an embedding. We use three convolutional layers in the submodel with ReLU activations. A global average pooling layer is used as a readout layer to aggregate the node representations into one graph embedding. These embeddings are stacked and passed through a fully connected layer for final classification (cf. Fig. 4.1).

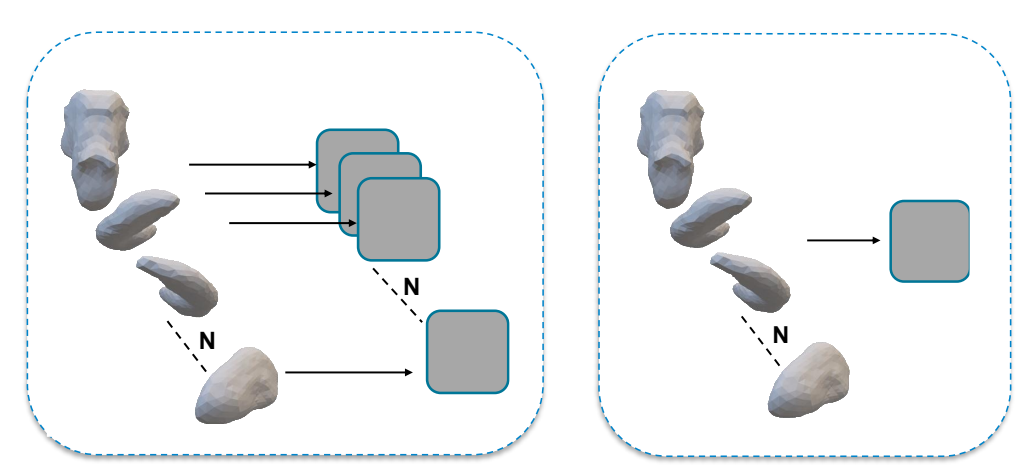


Figure 4.2: Non-shared submodel (left), shared submodel (right) Each sub-GNN learns structure-specific embedding. Shared GNN learns from all sub-graphs, generalising learnings across structures

4.3 Experiments

The experiments were designed to evaluate and compare three main aspects: (i) the choice of convolutional layers for the shared submodel; (ii) the choice of node features; (iii) the effect of data augmentation on robustness and generalisation, Figure 4.3.

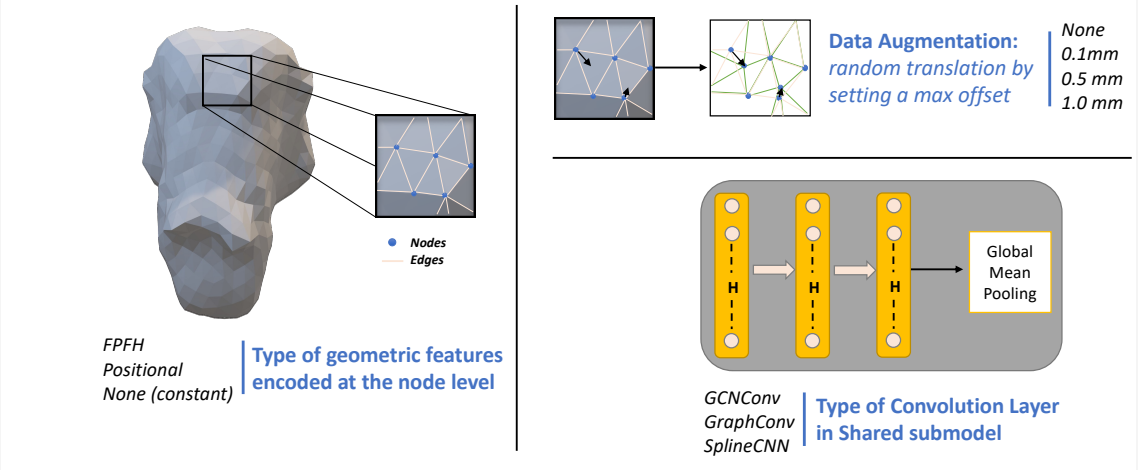


Figure 4.3: Aspects of comparison; choice of node features (left), data augmentation by perturbing the nodes (top-right) and type of convolutional layers (bottom-right)

4.3.1 Implementation and training

We use the Adam optimiser with a learning rate of 0.001 and the standard cross-entropy loss as the classification objective function. To increase the variability of the training data and to avoid overfitting, we employ a simple data augmentation strategy [244]. Individual graph nodes are randomly translated by a maximum offset. We evaluate the effect of the strength of augmentation and test maximum offsets of 0.1mm, 0.5mm, and 1.0mm, Fig. 4.3 top-right. Given the limited amount of training data, data augmentation should be beneficial for improving classification accuracy across different datasets.

All our implementations were done in PyTorch, benefiting from the excellent PyTorch Geometric library[1]. We use PyTorch Lightning[2] for ease of implementation of the model and data structures.

[1] https://pytorch-geometric.readthedocs.io/
[2] https://www.pytorchlightning.ai/

4.3.2 Task 1: Biological sex classification

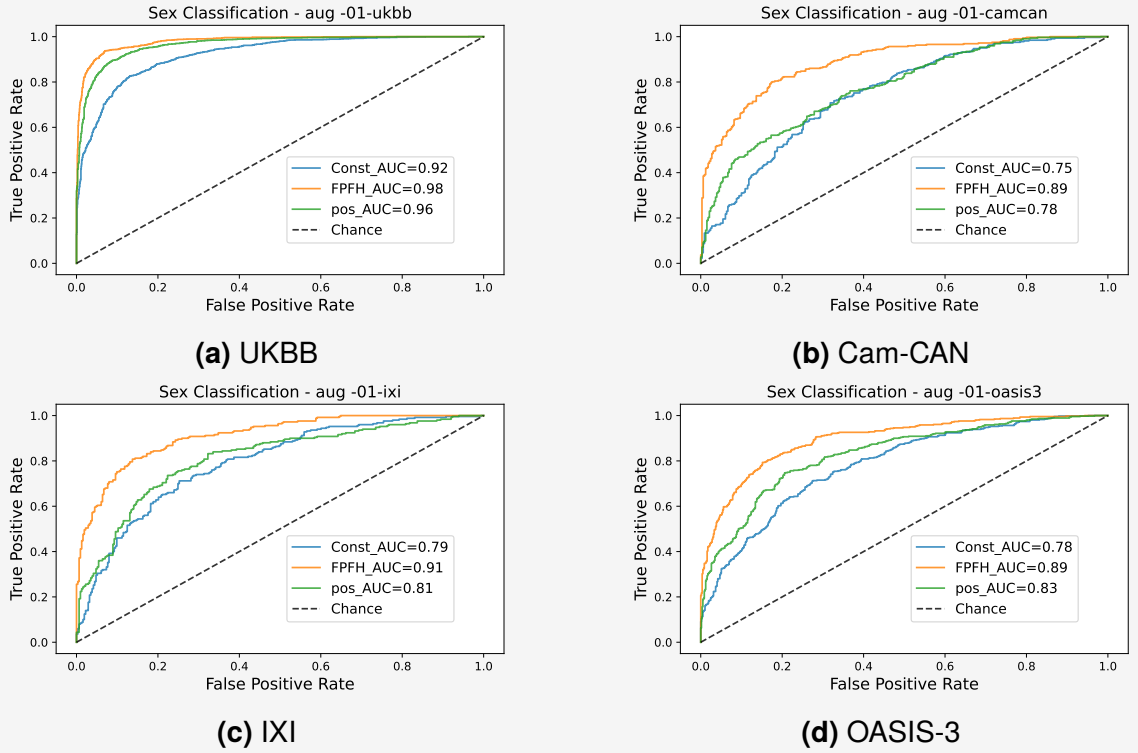
We use biological sex classification as a proof-of-concept task because several neuroimaging datasets from different sources are available for extensive testing. We use the UKBB data for the model development, with a data split of 70%, 10%, and 20% for training, validation, and testing. The batch size was set to 128, and all hidden features were set to 32 (both for convolutional and fully connected layers). When using SplineCNN, we set the kernel size to 5 and use the sum aggregation. The maximum number of training epochs was set to 50. Through iterative evaluation of training curves in TensorBoard, we determined 50 epochs as the maximum required for all models to converge, with some architectures stabilising earlier. We retained the model with the highest validation performance for the final evaluation on the test set.

Effect of node features To evaluate the effectiveness of different node features, in our first set of experiments, we employed SplineCNN in the shared convolutional submodel (as these performed well in the initial experimentation). We then evaluated classification performance on the UKBB test set, Cam-CAN, IXI, and OASIS-3 using constant, positional, and FPFH node features.

The ROC curves in Fig. 4.4 show that FPFH substantially outperforms other node features on all four datasets. It is worth noting that while positional features perform well on the in-distribution UKBB test set, these features underperform on out-of-distribution test sets. This is due to their reliance on Cartesian coordinates of mesh nodes, which do not generalise well due to differences in data acquisition. FPFH, on the other hand, are invariant to the pose of the mesh and show much better generalisation across datasets.

Effect of data augmentation Next, we evaluate the effect of varying strengths of data augmentation. The maximum offset for the random node translation varies from 0 (no augmentation) to 0.1, 0.5, and 1.0mm. The ROC curves in Fig. 4.5 demonstrate the benefit of data augmentation on robustness and generalisation. The best performance is achieved using data augmentation of 0.1, which increased AUC by 4-5% compared to not using augmentation. While data augmentation slightly decreases the performance on the in-distribution UKBB test set, it substantially improves performance on all out-of-distribution test sets, confirming the importance of adding random perturbations to the training data.

Effect of convolution layer Finally, we evaluate the three different convolutional layers, using FPFH as the node features and data augmentation of 0.1mm. In Fig. 4.6a, we observe similar performance for SplineCNN and GCNConv, closely followed



(a) UKBB **(b)** Cam-CAN **(c)** IXI **(d)** OASIS-3

Figure 4.4: ROC curves for sex classification comparing different node features across the datasets UKBB, Cam-CAN, IXI, OASIS-3 using SplineCNN in the submodel.

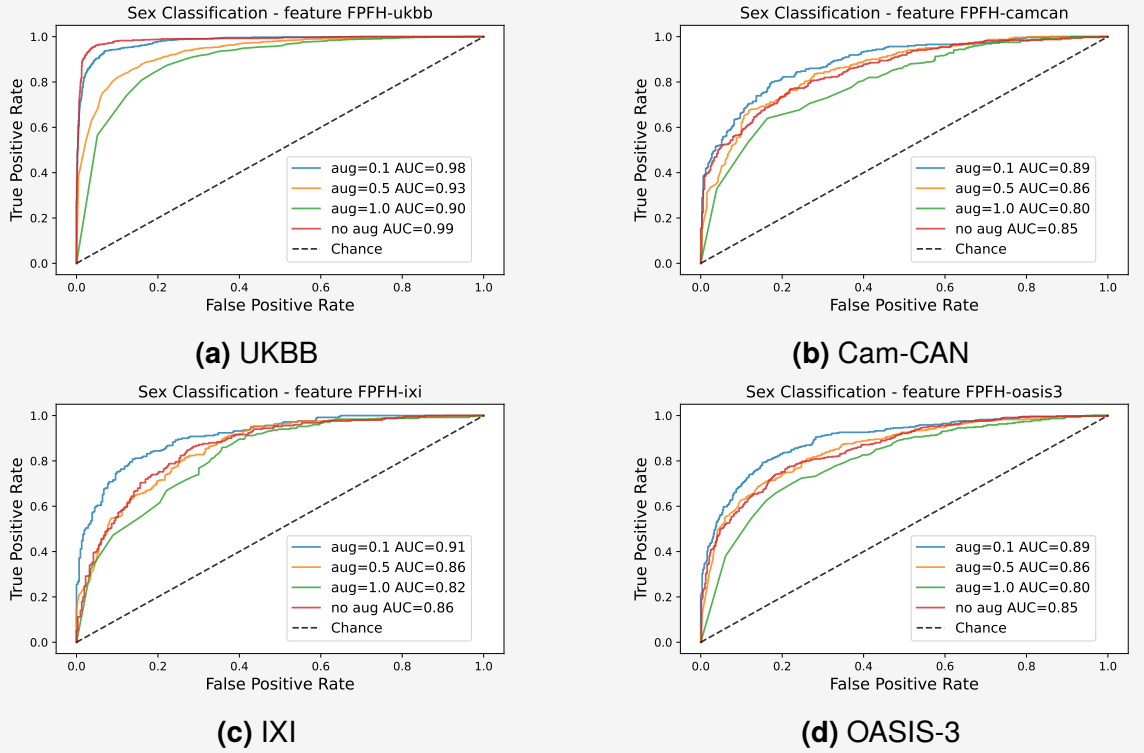
by GraphConv.

4.3.3 Task 2: Alzheimer's disease classification

To confirm whether the above findings hold for a clinically relevant task, we consider Alzheimer's disease (AD) classification on OASIS-3 with a 70%, 10%, and 20% train, validation, and test split. We evaluate the effect of the convolutional layer using a larger amount of data augmentation of 0.5mm due to the smaller amount of training data. We then also evaluate the effect of node features for AD classification, using SplineCNN in the submodel for consistency with the sex classification experiments. The results are shown in Fig. 4.6b and 4.6c. GCNConv performs slightly better than SplineCNN, with a substantially decreased performance for GraphConv. FPFH features again outperform other node features.

4.3.4 Bias analyses

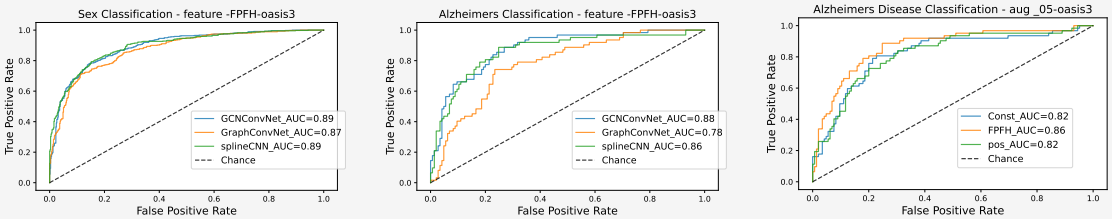
We also investigated potential biases in the predictions regarding subgroup performance disparities. To this end, we first analysed the biological sex classification model stratified by age groups. As the training data from UKBB only covers a limited age



(a) UKBB **(b)** Cam-CAN

(c) IXI **(d)** OASIS-3

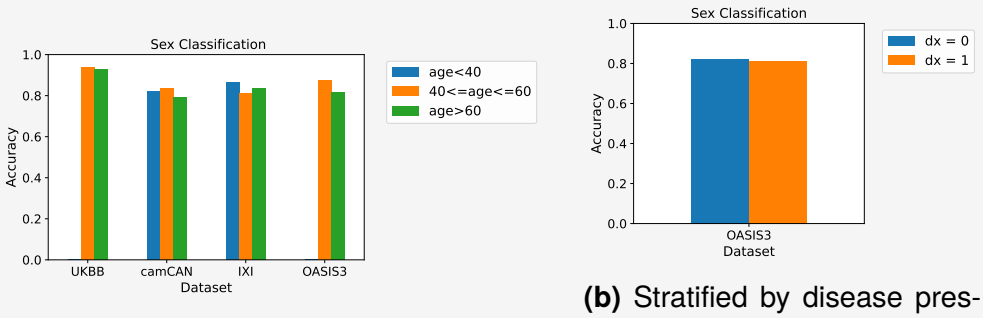
Figure 4.5: ROC curves showing the effect of data augmentation for sex classification across domains, using SplineCNN as the shared submodel and FPFH as node features.



(a) Sex - Conv. Layers **(b)** AD - Conv. Layers **(c)** AD - Node Features

Figure 4.6: Effect of convolution layer for (a) sex and (b) Alzheimer's disease classification. (c) Effect of node features on AD classification. All were evaluated on OASIS-3.

range between 44 and 73-year-old subjects, we wanted to understand whether the performance might degrade for younger subjects. However, the results shown in Fig. 4.7a suggest that the sex classification model with SplineCNN and FPFH features generalises well across the entire age range. Both Cam-CAN and IXI contain many subjects in the range of 18 to 40 years. Next, we analysed whether sex classification may be affected by disease status. Here, we looked at the classification performance separately for the group of healthy controls and subjects with Alzheimer's disease. Again, we find no differences in the classification accuracy, suggesting that the sex classification model generalises well (cf. Fig. 4.7b).



(a) Stratified by age groups

(b) Stratified by disease presence

Figure 4.7: Bias analysis for sex classification using SplineCNN with FPFH features. Classification performance is stratified by (a) age groups and (b) the presence of disease.

4.4 Conclusion

This comparative study evaluated the effect of node features, convolutional layers, and data augmentation on two different tasks and four datasets in medical shape classification with graph neural networks. We find that the use of FPFH features is highly beneficial, substantially improving classification performance on out-of-distribution test data. The FPFH features alleviate the need for data normalisation, such as mesh alignment, due to their pose invariance. We are not aware of earlier studies proposing the use of FPFH in GNNs. We further find that SplineCNN and GCNConv are both viable options for the convolutional layers, yielding comparable performance. We also conclude that data augmentation is essential in GNNs, in particular, when the amount of training data is limited. We find that stronger data augmentation is beneficial, in particular for Alzheimer's disease classification, where the training set contained less than 800 samples. AD classification performance was overall promising, and in line with a recent study evaluating data representations [177]. This should be confirmed in future work on other datasets for AD classification, such as ADNI.

A limitation of this work is that only relatively simple data augmentation was considered in the form of perturbing the mesh node positions. Recently, there has been work on more advanced data augmentation techniques for graph neural networks [47], such as removing a certain number of edges either randomly [171] or based on the GNN predictions, adjusting the graph in an adaptive manner [28], to name a few. In future work, it would be interesting to evaluate the effect of these more advanced techniques on medical shape classification.

Chapter 5

The Importance of Model Inspection for Better Understanding Performance Characteristics of Graph Neural Networks

This chapter extends the comparative performance analyses from chapter 4 by highlighting the importance of conducting a comprehensive model inspection. Here, we investigate the effect of modelling choices on the feature learning characteristics of GNNs applied to a brain shape classification task. Specifically, we analyse the effect of using parameter-efficient, shared graph convolutional submodels compared to structure-specific, non-shared submodels. Further, we assess the effect of mesh registration as part of the data harmonisation pipeline. We find substantial differences in the feature embeddings at different layers of the models. Our results highlight that test accuracy alone cannot identify important model characteristics such as encoded biases related to the data source or potentially non-discriminative features learned in submodels. Our model inspection framework offers a valuable tool for practitioners to better understand the performance characteristics of deep learning models in medical imaging.

This chapter is based on the following publication:

> *Shehata, N., Pic¸arra, C., Kazi, A., et al. "The Importance of Model Inspection for Better Understanding Performance Characteristics of Graph Neural Networks". In: 2024 IEEE International Symposium on Biomedical Imaging (ISBI). 2024, pp. 1–5. DOI: 10.1109/ISBI56570.2024.10635526.*

5.1 Introduction

Understanding biological sex-based differences in brain anatomy provides valuable insights into both neurodevelopmental processes and cognitive functioning. Recent strides in the field of geometric deep learning [19], particularly the advent of Graph Neural Networks (GNNs), have revolutionised the analysis of complex, non-Euclidean data [222] to make predictions at a node, edge, or graph-level. This allows us to treat brain shapes as graphs, leveraging the power of GNNs to learn from complex structural anatomical data [177]. Discriminative feature embeddings can be withdrawn from these models, representing brain shapes as a continuous vector of numerical features that capture valuable structural and geometrical information for downstream prediction tasks [14]. Techniques like Principal Component Analysis (PCA) can be used to reduce the dimensionality of graph embeddings for visualisation, aiding the exploration of subgroup biases in the feature space beyond the target label. This analysis may help practitioners ensure the reliability of their predictions and is particularly important in applications where GNNs feature embeddings may be leveraged for new tasks, such as fine-tuning, domain transfer, or multi-modal approaches.

In this study, we dissect GNN models trained under different settings for the task of sex classification using 3D meshes of segmented brain structures. We inspect the learned feature embeddings at different layers within a multi-graph neural network architecture. Through this granular analysis, we reveal critical insights into the inner workings of our models, identifying important effects of different modelling choices. This research demonstrates the utility of conducting a model inspection framework as part of model development, highlighting insights that may guide practitioners in the selection of models with desired characteristics, avoiding biases, overfitting and better understanding the driving forces behind predictions.

5.2 Methods

5.2.1 Multi-graph neural network architecture

We use the same model as described in chapter 4, which is comprised of two main components; the *GCN submodel* which aims to learn graph embeddings over 3D meshes using multiple graph convolutional layers and an *MLP classification head* that takes the graph embeddings as inputs and performs the final classification using two fully connected layers (cf. Fig. 5.1).

The input to our models is 15 subgraphs representing 15 brain structures, extracted from T1-weighted brain scans. We consider two approaches for learning graph embeddings with GCN submodels, either using a shared submodel or non-shared (please refer to section 4.2.2 for details on shared vs non-shared submodels).

As before, in both approaches, the architecture of the GCN submodel is identical and consists of three graph convolutional layers [109] with Rectified Linear Unit (ReLU) activations. A global average pooling layer is used as a readout layer, aggregating node representations into a single graph-level feature embedding. The embeddings from individual structures are stacked to form a subject-level feature embedding which is passed to the classification head.

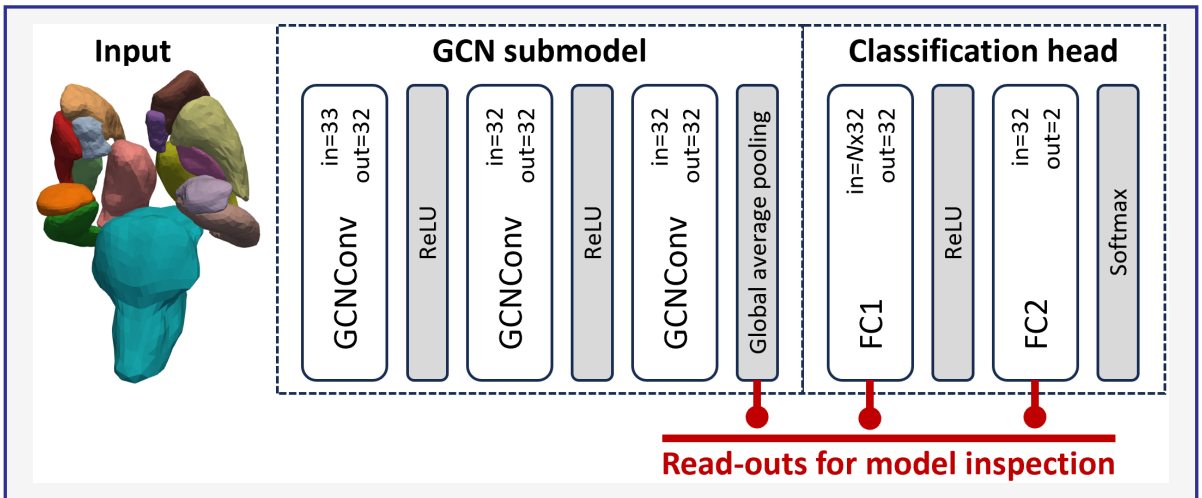


Figure 5.1: Model architecture consisting of a graph convolutional network (GCN) submodel feeding graph embeddings into a classification head with two fully connected layers (FC1 and FC2), where N is the number of brain substructures, 15. For our model inspection, we read out the feature vectors from the GCN submodel, FC1, and FC2.

5.2.2 Model inspection

Our model inspection approach is focused on evaluating the separability of the target label (biological sex, Male and Female) and data source classes (UKBB, CamCAN, IXI or OASIS3) through feature inspection - datasets detailed in chapter 3. Each test set sample is passed through the complete pipeline and its feature embeddings are saved at three different stages: at the output layer of the GCN submodel and at the first (FC1) and second (FC2) fully connected layers of the classification head. The dimensions of these embeddings are, respectively, 480 (15 substructures times the hidden layer size, 32), 32 and 2. To allow for visual inspection, the feature embeddings from the GCN and FC1 layers are inputted to a PCA model to reduce their dimensionality. The PCA modes capture the directions of the largest variation in the high-dimensional feature space, allowing us to visualise feature separation in 2D scatter plots. We randomly sample 500 subjects from each dataset for the visualisations. Given that all the models were trained to classify biological sex, a clear separation should be expected between the Male and Female classes in the first PCA modes.

5.3 Experiments & Results

For a thorough evaluation, we trained and tested the four models - shared and non-shared GCN submodels, and with and without mesh rigid registration - on identical data splits. All code was developed using PyTorch Geometric and PyTorch Lightning for model implementation and data handling. We used the Adam optimiser [108] with a learning rate of 0.001, and employed the standard cross-entropy loss for classification. Random node translation was used as a data augmentation strategy with a maximum offset of 0.1mm [244]. This was shown to improve performance in our previous study from chapter 4. Model selection was done based on the loss of the validation set. Our code is made publicly available[1].

5.3.1 Classification performance

Figure 5.2 summarises the classification performance of the four models, showing the ROC curves and the area under the curve (AUC) metric, reported separately for each of the four test datasets. There are two main observations: (i) There are very few differences in the absolute performance across the four models. Comparing the *shared* vs *non-shared* submodel, the AUC performance is comparable. When comparing models *with* and *without* mesh registration, we find the generalisation gap decreases between the in-distribution test (UKBB) and the external test data (CamCAN, IXI, OASIS3). However, we also observe a small drop in performance on the in-distribution test data when using mesh registration, compared to not using registration. A practitioner using internal test results for final model selection may opt for using a *shared* submodel due to its parameter efficiency, *without* mesh registration due to convenience. As we will see next, this choice may be suboptimal as test accuracy alone is insufficient to identify important model characteristics.

The similar performance between the four models, suggests that global shape patterns rather than fine-grained local correspondences primarily drive the classification task. The GNNs capture relational and structural information robust to small local misalignments, particularly when labels correlate with global shape. Shared weights act as implicit regularisation, focusing learning on consistent patterns across datasets. Mesh registration, while aligning fine-grained correspondences, adds limited benefit when the predictive signal is embedded in global shape differences, explaining the similar performance observed across different configurations.

[1] https://github.com/biomedia-mira/medmesh

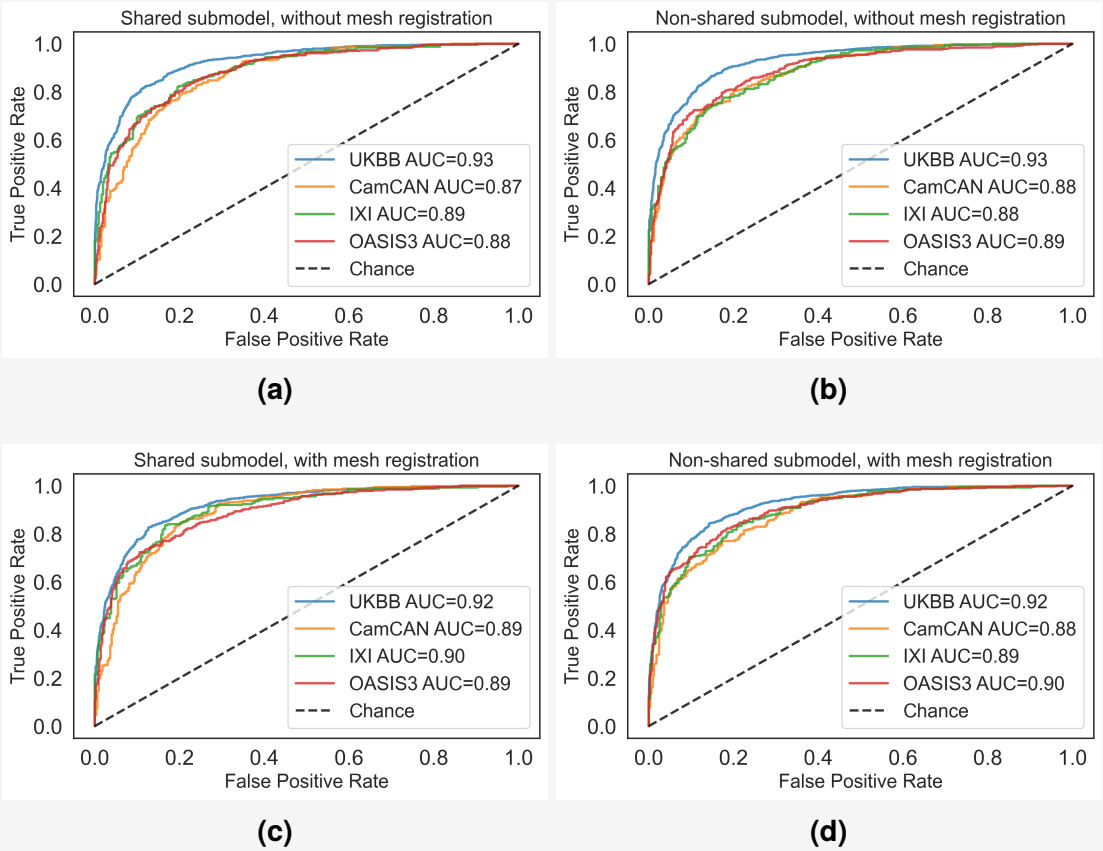


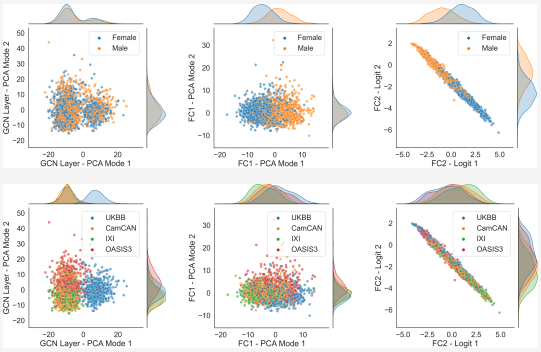
Figure 5.2: Sex classification performance for four models; **(a)** *shared* and **(b)** *non-shared* submodel *without* mesh registration, **(c)** *shared* and **(d)** *non-shared* submodel *with* mesh registration. We observe that the generalisation gap between the in-distribution test data (UKBB) and the external test data (CamCAN, IXI, OASIS3) closes *with* mesh registration. Overall, there are only small differences in performance, illustrating that test accuracy alone cannot identify variations in model characteristics.

5.3.2 Effect of using structure-specific submodels

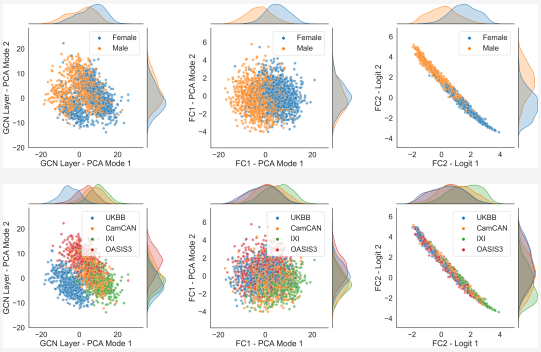
For the models that use a *shared* submodel, we observe that the GCN feature embeddings are non-discriminative with respect to the target label. Separation seems completely missing in the *shared* model *without* registration (see Fig. 5.3a), with only weak separation in the *shared* model *with* registration (see Fig. 5.3c). The classification heads will primarily contribute to the performance of these models. For the models with a *non-shared* submodel, we find a much better separability for the GCN features *with* and *without* mesh registration (cf. Figs. 5.3b, d). Here, the GCN features will meaningfully contribute to the models' classification performance.

5.3.3 Effect of mesh registration

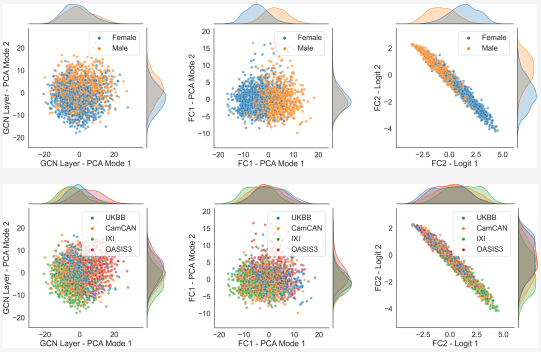
When studying the effect of mesh registration, we can clearly observe that *without* registration, the GCN feature embeddings from the submodel strongly encode data source, showing separate clusters for UKBB and external test data (cf. Figs. 5.3a,b). When introducing mesh registration as a pre-processing step, we note a significant improvement, with an almost entirely removed separation of datasets in the GCN layer independent of whether a *shared* and *non-shared* submodel is used (Figs. 5.3c, d). The separability of the target label in the GCN layer is well defined for the *non-shared* submodel (Fig. 5.3d), while remaining weak for the *shared* submodel (Fig. 5.3c). Rigid registration as a pre-processing step seems to not only improve the learning efficiency of the GCN submodel, but also its ability to generalise across data distributions.



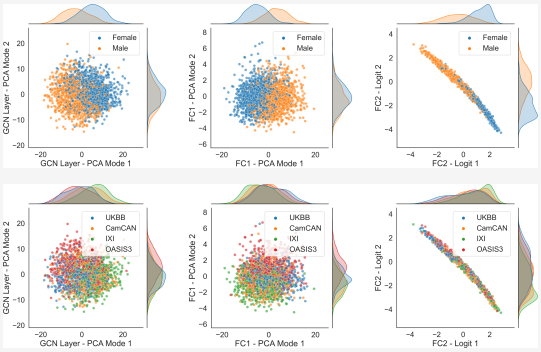
(a) *Shared* submodel, *without* mesh registration



(b) *Non-shared* submodel, *without* mesh registration



(c) *Shared* submodel, *with* mesh registration



(d) *Non-shared* submodel, *with* mesh registration

Figure 5.3: Effect of modelling choices on feature separability for four different models at their GCN layer (left), first fully connected layer FC1 (middle), and output layer FC2 (right). **Models**: **(a,c)** *shared* and **(b,d)** *non-shared* GCN submodel, and **(a,b)** *without* and **(c,d)** *with* mesh registration. For each model, we show the separation by target label in the top row and the separation by dataset in the bottom row. **Effect of submodel:** The models in **(a,c)** with a *shared* submodel are unable to learn discriminative GCN features for the prediction task, while the models in **(b,d)** with a *non-shared* submodel show much better task-related separability in the GCN features. **Effect of registration:** The models in **(a,b)** *without* registration strongly encode information about the data source in the GCN layer. This is much reduced for the models in **(c,d)** *with* mesh registration.

5.4 Conclusion

Our findings underscore the limitations of relying solely on test accuracy for model selection, particularly when focusing on in-distribution test accuracy. We demonstrate that this may lead practitioners to select models with undesired characteristics where GCN features are non-discriminative for the prediction task and/or strongly encode biases such as data source. Using a comprehensive model inspection, we were able to identify variations in the model characteristics and better understand what drives the final prediction (GCN submodel vs classification head). The importance of this becomes evident when considering applications such as fine-tuning, domain transfer, or multi-modal approaches, where GCN feature embeddings may be leveraged for new tasks.

Our model inspection framework can be easily applied to other models, tasks, and purposes. It was previously used to detect biases in chest radiography disease detection models [68]. Here, we strongly advocate for the wider use of model inspection as an integral part of comparative performance analyses. We hope that our work can contribute to improving the reliability of model selection in all areas of deep learning for biomedical image analysis.

Chapter 6

Combining Imaging and Shape Features for Prediction Tasks of Alzheimer's Disease Classification and Brain Age Regression

We investigate combining imaging and shape features extracted from MRI for the clinically relevant tasks of brain age prediction and Alzheimer's disease classification. Our proposed model fuses ResNet-extracted image embeddings with shape embeddings from a bespoke graph neural network. The shape embeddings are derived from surface meshes of 15 brain structures, capturing detailed geometric information. Combined with the appearance features from T1-weighted images, we observe improvements in the prediction performance on both tasks, with substantial gains for classification. We evaluate the model using public datasets, including CamCAN, IXI, and OASIS3, demonstrating the effectiveness of fusing imaging and shape features for brain analysis.

This chapter is based on the following publication, presented as an oral presentation at UK AI Conference 2025:

Shehata N, Piçarra C, Glocker B. Combining imaging and shape features for prediction tasks of Alzheimer's disease classification and brain age regression. arXiv preprint arXiv:2501.07994. 2025 Jan 14.

6.1 Introduction

Alzheimer's disease (AD) is a progressive neurodegenerative disorder, leading to substantial memory impairment, making early diagnosis crucial [22]. Additionally, predicting brain age using MRI has become a clinically important method for identifying imaging biomarkers associated with various neuropathologies. Many studies have proposed brain age prediction as a means to characterise these conditions [36].

Neuroimaging, such as structural MRI, has become indispensable for diagnosis. With the availability of large-scale datasets, there has been a surge in applying sophisticated machine learning algorithms to enhance our understanding and prediction capabilities of neurodegenerative disease [61, 137, 234]. The complex anatomy of brain structures necessitates sophisticated approaches to fully capture spatial relationships and geometric properties. The advancement of automated segmentation techniques [137], facilitates anatomical shape extraction from medical images. Extracted shapes can then be represented as meshes, enabling the use of powerful GNNs for learning geometric features. Graph neural networks (GNNs) have been increasingly applied for disease prediction from neuroimaging data, including Alzheimer's disease, demonstrating the ability to model connectivity and geometric structure effectively [3, 14, 79, 146, 248].

CNNs have shown remarkable success in extracting meaningful features from structural MRI, contributing to regression, classification and image segmentation tasks. Notable architectures such as ResNet, VGG and AlexNet have been particularly effective in these applications [137]. The increasing complexity and richness of neuroimaging data have driven the need to utilise the strengths of different imaging modalities, leading to the development of multimodal fusion models [121]. Recent work demonstrated promising results fusing imaging and non-imaging features such as demographics, lifestyle and cognitive performance [16]. Traditionally, multimodal fusion models have combined different imaging modalities (e.g., T1+T2 MRI, MRI+PET) [131], and ensemble-based within-modality fusion has also been explored, where multiple deep learning models trained on the same modality (e.g., T1-weighted MRI) are combined to enhance predictive performance for Alzheimer's disease and MCI diagnosis [213]. Recent studies have proposed multi-modal GNN frameworks that integrate imaging, clinical, and genetic data for AD diagnosis, demonstrating improved predictive power and interpretability [3, 207, 228, 243]. Within-modality representation fusion using different data representations from the same MRI modality, such as combining CNN features from multiple 3D angular orientations, has also shown promise for enhancing AD classification [207]. However, the fusion of CNN-derived image em-

beddings with GNN-extracted geometric shape embeddings—both derived from the same T1-weighted MRI—for brain age prediction and AD classification remains limited. Brain age prediction has progressed from using handcrafted volumetric features and cortical thickness [59, 62] to CNN-based pipelines that learn complex imaging patterns [11, 36, 41, 99, 197], with multimodal MRI approaches combining structural, diffusion, and functional imaging improving predictive power [123]. Similarly, AD classification literature spans traditional ML approaches with engineered features [161] to advanced CNN methods on MRI [7, 141] and multimodal pipelines combining MRI, PET, and clinical data [115, 194, 236].

In this chapter, we present a deep learning approach that integrates imaging features with brain shape features extracted from structural MRI to enhance prediction accuracy for brain age regression and AD classification. Our approach takes advantage of the strengths of both image-based and graph-based DL techniques. Specifically, we employ a ResNet-18 for extracting features from the T1-weighted MRI and a multi-graph neural network architecture for processing brain surface meshes. By combining these complementary feature embeddings, our model aims to capture a comprehensive representation of brain anatomy, potentially improving predictive performance compared to models using either imaging or shape features alone. Our results demonstrate that the fusion model outperforms baseline models, highlighting the potential of combining image-based and graph-based features for more accurate predictions requiring only standard T1-weighted scans.

6.2 Imaging datasets and brain shape representation

We use three public datasets; CamCAN, IXI and OASIS3; please see sections 3.1.2, 3.1.3 and 3.1.4, respectively for more details on acquisition and section 3.2 for the postprocessing pipeline applied by which 15 surface meshes are extracted per subject from the MR images.

The extracted brain shapes are represented as graphs as described in 3.3, employing Fast Point Feature Histograms (FPFH) feature descriptors as node features (more on FPFH in section 3.3) because this has been proven to boost our bespoke model performance as shown in chapter 4.

All meshes are aligned to a structure-specific standard orientation by performing rigid registration using the closed-form Umeyama method [206], detailed in section 2.6.1. One subject is selected randomly from the dataset serving as the reference mesh.

6.3 Feature extraction & fusion

Our feature extraction process employs two distinct models: an image-based model using 3D adaptation of ResNet-18 architecture and a graph-based model utilising a bespoke multi-graph architecture described below. Fig. 6.1 provides an overview of our fusion model, illustrating how the two feature extraction models are integrated. The readouts from the pooling layer of the graph-based model and from the second-to-last layer of the ResNet are concatenated and passed to a linear predictor.

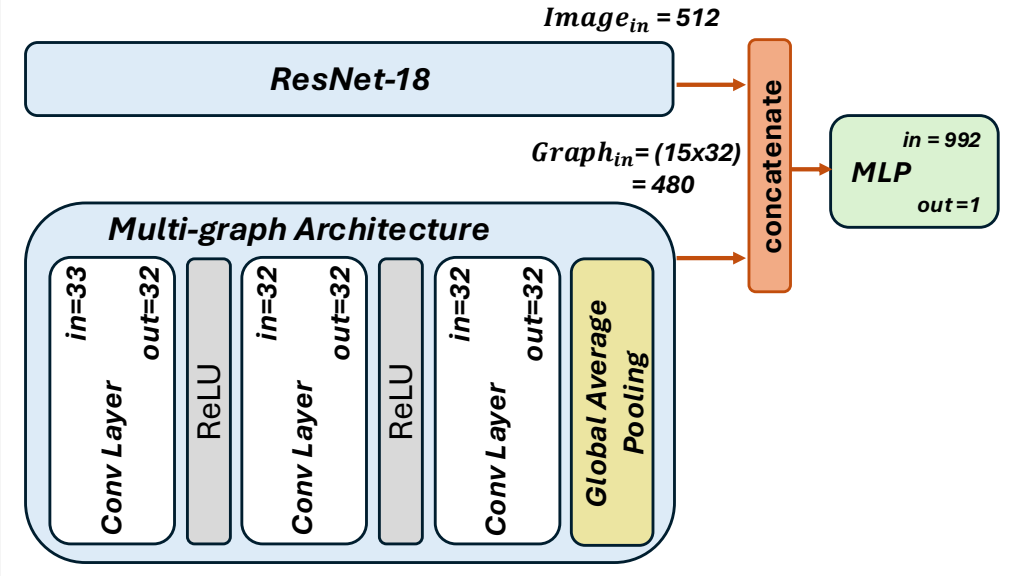


Figure 6.1: Fusion model overview, consisting of image-based and graph-based feature extractors, embeddings concatenated and passed through a multi-layer perceptron (MLP).

Imaging features. We use ResNet-18 from PyTorch [148], with the last linear layer removed after training from scratch on our training sets. Data augmentation consists of horizontal flip, gamma intensity adjustment, the addition of Gaussian noise with random parameters, and the application of a random affine transformation.

Shape features. To utilise all 15 brain shapes, a multi-graph architecture is designed with a submodel for each shape. The embeddings are concatenated and fed into two FC layers. The submodels have three convolutional layers with Rectified Linear Unit (ReLU) activations and a global average pooling layer to aggregate shape representations [185]. Using structure-specific submodels was previously found to yield more discriminative shape features [186].

6.4 Experiments & Results

The same samples per data split in training, validation, and test sets were used consistently for the image-based, graph-based models, and fusion models, ensuring a fair comparison between all approaches. Implementations of the models and data handling are based on PyTorch Geometric [1] and PyTorch Lightning [2]. We used the Adam optimiser with a learning rate of 0.001 and employed the standard cross-entropy loss for AD classification and mean squared error for brain age regression.

Random node translations with a maximum strength of 0.1mm were applied as data augmentation when training the graph-based models [244]. For the graph convolutional layers, we compare SplineConv [57] and GCNConv [109] (also detailed in section 4.2).

We extract the feature embeddings from three different models; the image-based model (ResNet) and two graph-based models (SplineConv and GCNConv). The embeddings are passed through an MLP independently or concatenated for the fusion models. Three random seeds were used to calculate the mean and standard deviation of the performance.

6.4.1 Brain age regression

We split the combined dataset of CamCAN and IXI into 875 training, 97 validation and 243 testing samples. The age range is 18-88 years. The models are evaluated using mean absolute error (MAE), and R2 scores. Table 6.1 shows the results over three runs for five different models. The scatter plots in Fig. 6.2 visualise the relationship between true and predicted ages.

Model	MAE ↓	R2 score ↑
ResNet	4.391±0.005	0.889±0.000
SplineConv	7.151±0.027	0.724±0.007
GCNConv	6.734±0.035	0.741±0.005
Fusion w/ SplineConv	**4.362±0.074**	**0.893±0.004**
Fusion w/ GCNConv	4.388±0.037	0.890±0.001

Table 6.1: Evaluation metrics for brain age regression using embeddings extracted from image- and graph-based models.

[1] https://pytorch-geometric.readthedocs.io/
[2] https://lightning.ai/docs/pytorch/stable/

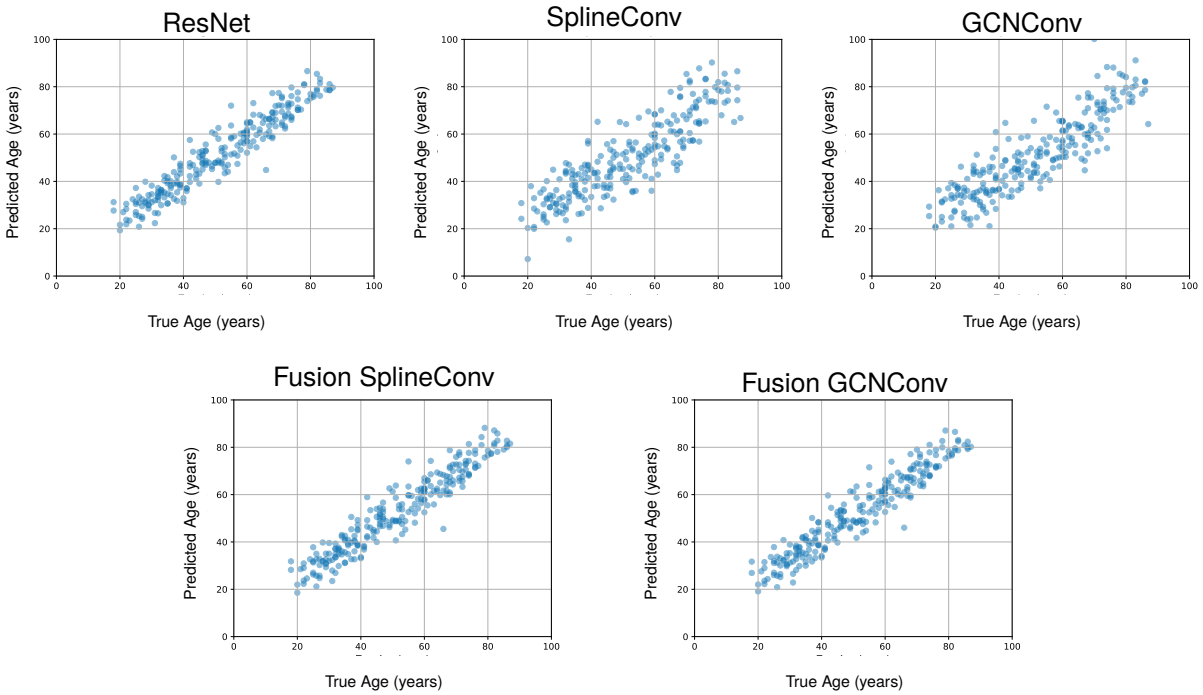


Figure 6.2: Scatter plots of true age against predicted age for different brain age prediction models.

6.4.2 Alzheimer's disease classification

OASIS-3 was split into 745, 82, and 207 for training, validation, and testing, respectively. The samples per class for the test set were 145 and 62.

Results are shown in Table 6.2 for the five models; the first three are from passing the embeddings extracted from the baseline models through an MLP and the last two are from concatenating the embeddings into an MLP. A single run for each model was selected to plot the ROC curves to better compare the performance of passing the extracted features from the standalone models with and without concatenating them through an MLP, Fig. 6.3.

Model	AUC ↑	TPR@FPR=0.15 ↑	TPR@FPR=0.20 ↑	FPR@TPR=0.70 ↓
ResNet	0.848 ± 0.018	0.656 ± 0.042	0.715 ± 0.050	0.172 ± 0.046
SplineConv	0.791 ± 0.005	0.651 ± 0.015	0.742 ± 0.013	0.162 ± 0.020
GCNConv	0.837 ± 0.019	0.629 ± 0.013	0.715 ± 0.042	0.192 ± 0.046
Fusion w/ SplineConv	0.817 ± 0.008	0.688 ± 0.020	0.758 ± 0.000	0.152 ± 0.020
Fusion w/ GCNConv	**0.861 ± 0.007**	**0.742 ± 0.013**	**0.780 ± 0.027**	**0.121 ± 0.008**

Table 6.2: AD classification results from using embeddings from image- and graph-based models compared to fusing embeddings from both. We report the AUC, TPR@FPR and FPR@TPR at varying performance levels. We observe that using fused feature embeddings outperforms standalone models consistently across TPR/FPR performance levels.

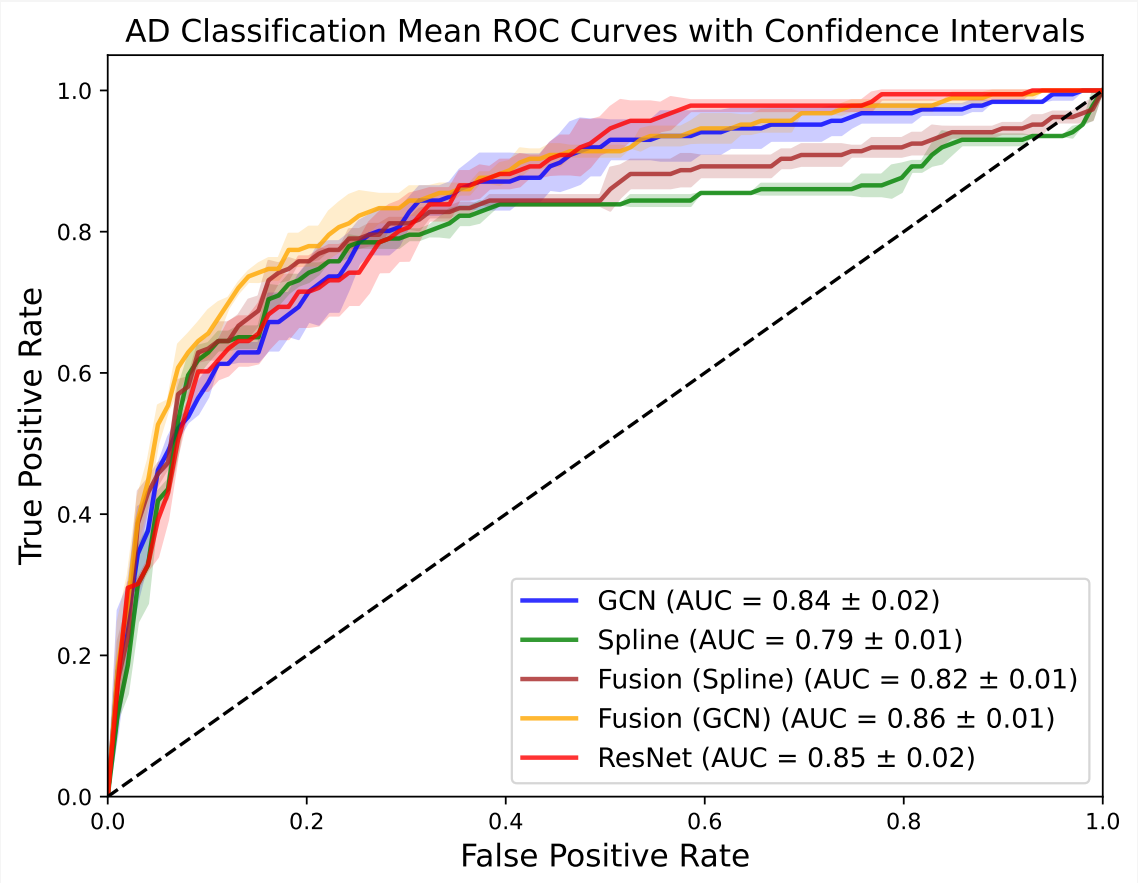


Figure 6.3: AD classification results from using embeddings from baseline models, passing them through the MLP (one FC).

6.5 Discussion

For brain age regression, we observe some improvement with fusion models, but it remains unclear whether fusion truly adds significant value over standalone models. The graph-based models, which extract shape features, may offer complementary information to the image-based model, but their impact is limited when combined for age prediction (cf. Table 6.1). While in standalone, we observe better performance for GCNConv, a slight improvement is observed for the fusion model when using SplineConv. This observation suggests that better standalone performance may not always translate to better performance in a fusion model.

In contrast, we observe a substantial improvement with fusion models for AD classification, suggesting that the fusion of imaging and shape features is beneficial for this more challenging classification task. The results in Table 6.2 indicate that combining the features leads to better performance in the critical regions of lower FPR, which is particularly important for clinical applications. Interestingly, using AUC alone does not fully reveal these improvements. While AUC values for the ResNet and fusion models appear similar (0.848 vs 0.861), a closer inspection of the ROC curves shows noticeable performance differences in the clinically relevant areas of low FPR. Additionally, by evaluating TPR at fixed FPR values and FPR at fixed TPR, we see substantial improvements in AD classification performance using fusion models. Emphasising that these metrics provide a more detailed understanding of the model's behaviour in the regions of the ROC curve that are most important for practical use. This demonstrates that the fusion of imaging and shape features consistently leads to more reliable AD classification.

The observation that SplineConv outperformed GCNConv for brain age regression, while GCNConv was superior for AD classification, likely relates to their mathematical properties and the nature of the prediction tasks. SplineConv uses B-spline kernels, enabling smoother, continuous aggregation suitable for capturing gradual, distributed changes associated with ageing [57]. Whereas, GCNConv uses simpler, isotropic neighbourhood aggregation, favouring sharp, localised feature learning aligned with the focal atrophy patterns characteristic of AD [109]. Clinically, brain ageing reflects diffuse, global changes [166], whereas AD-related changes (e.g., hippocampal atrophy) are more abrupt and localised [96]. The choice of graph convolution should thus align with the underlying biological processes and task objectives, guiding architecture selection for clinical deployment.

6.6 Conclusion

By leveraging deep learning architectures tailored for different data representations, images and shapes, we find substantially improved predictive accuracy for AD classification. Importantly, evaluating models for AD classification using ROC curves rather than AUC alone reveals performance differences at clinically relevant classification thresholds.

Future work will evaluate the model on larger and more diverse datasets and other prediction tasks. Additionally, we plan to explore using different image modalities, such as functional MRI, to enhance the model's performance further. While our simple approach of concatenating imaging and shape features showed improvements, more sophisticated fusion methods, such as attention mechanisms or adaptive fusion, could be beneficial. We also plan to interpret the contribution of each modality to the final predictions. This could provide valuable insights into the structural changes associated with brain ageing and AD progression.

This work may also inspire the use of different data representations in domains other than neuroimaging. For example, prediction tasks such as the malignancy of lung nodules could benefit from the fusion of imaging and shape features.

Chapter 7

Limitations of GNN explainability methods: A case study on sex classification from brain shapes

After understanding the model characteristics in Chapter 5, the next step is to explore how the model arrives at its decisions because it can be just as important as the accuracy of the decision itself, specifically for validation and clinical transparency. To do so, it is important to identify which subgraphs of the input graph significantly influence the model's predictions. This study investigates the limitations of Graph Neural Networks (GNNs) explainability in brain shape analysis, focusing on the example of a multi-graph network trained on brain shapes for sex classification. We compare two approaches, GNNExplainer and a custom perturbation-based method, to identify influential brain substructures.

We propose a calibration-aware framework that integrates multi-level explanations, combining global substructure analysis with node-specific insights to bridge granular and systemic interpretability. Our experiments highlight key limitations in current GNN explainability methods, emphasise the biological relevance of identified substructures, and underscore the need for clinical validation to ensure reliable application in real-world settings.

Some of the work from this chapter contributed to:

> *Inácio MH, Shah M, Jafari M, Shehata N, Meng Q, Bai W, Gandy A, Glocker B, O'Regan DP. Cardiac age prediction using graph neural networks. MedRxiv. 2023 Apr 20:2023-04.*

7.1 Introduction

A key limitation of deep models and Graph Neural Networks (GNNs) is that they are not easily interpretable [105, 231]. Understanding how a model makes decisions is crucial in medical imaging to ensure transparency, validate predictions, and identify clinically meaningful biomarkers [29]. This is important to ensure the model makes the right prediction and is based on some understanding to make it robust and generalizable. In medical imaging, understanding these models can reveal undiscovered phenotypical biomarkers to clinicians. This also enhances the model's transparency and understandability, increasing trust for clinical deployment.

To address the challenge of providing explanations for GNN predictions, several frameworks have emerged which can be classified as gradient-based or graph-specific:

Gradient-based methods aim to quantify the importance of individual features influencing the model's output. They are also model-aware, relying on internal model parameters to identify feature contributions, revealing the relationship between the input features and output predictions. The most prominent gradient-based methods are saliency [190] and Integrated Gradients [195].

Graph-specific methods can be further grouped as perturbation-based, surrogate models or counterfactual explanations.

- Perturbation-based methods such as GNNExplainer [229] and others [130, 178, 215, 223, 232] perturb the graph structure or features and measure the impact on model predictions.
- Surrogate models [49, 87, 158, 212, 240] build interpretable models to approximate the GNN's predictions.
- Counterfactual explanations [33, 97, 182] generate alternate scenarios to explain predictions by identifying minimal changes to flip the outcome.

In addition to these three groups providing instance-level explanations, some methods focus on global model behaviour [9, 216, 230].

All these methods use post-hoc approaches, meaning they analyse the predictions of pre-trained GNNs retrospectively by training auxiliary models or applying perturbations, which may fail to fully capture the original GNN's decision-making process, leading to incomplete or unreliable explanations [39, 89]. Furthermore, they prioritise node- or edge-level importance (neglecting broader structural dependencies inherent to biological systems), conflate structural size with importance (disproportionately highlighting larger areas while overlooking smaller, clinically relevant areas), and focus

on instance-level explanations or global patterns (failing to integrate both).

This chapter addresses these gaps by providing a more comprehensive and reliable GNN explainability framework specifically tailored to our bespoke multi-graph architecture. To solve these issues, we propose a perturbation-based method tailored to the multi-graph architecture, enabling evaluation of the systemic impact of entire subgraphs rather than isolated nodes and identifying critical subgraphs that node-centric approaches overlook. The inconsistent rankings of critical substructures between our method and GNNExplainer underscore how existing approaches fail to capture biologically meaningful dependencies, validating our critique and emphasising the need for clinical validation. We reduce bias related to size by adjusting GNNExplainer's importance scores based on the volume of substructures, which helps us identify meaningful patterns regardless of size. We bridge the gap between instance-level and global analyses by using complementary evaluations of GNNExplainer importance scores: (1) across-subject analyses to identify population-level biomarkers and ensure generalizability, and (2) per-substructure analyses to pinpoint localised drivers, preserving anatomical specificity. Furthermore, we reduce bias related to size by adjusting GNNExplainer's importance scores based on the volume of substructures, which helps us identify meaningful patterns regardless of size. The inconsistent rankings of critical substructures between our method and GNNExplainer underscore how existing approaches fail to capture biologically meaningful dependencies, validating our critique and emphasising the need for clinical validation.

7.2 Methods

We employ a perturbation-based method to assess the systemic impact of altering entire subgraphs, providing a complementary perspective to node-centric explanations. For node-centric explanations, we use GNNExplainer [229] to compute node importance scores, highlighting individual nodes' influence on the model's predictions. We explore two key configurations in GNNExplainer: model explanation, which reveals how the architecture makes predictions, and phenomenon explanation, which uncovers the underlying patterns driving the predictive tasks. We propose two methods for analysing the scores—across subjects and within a subgraph—to determine which brain substructure is most important. We introduce additional qualitative and quantitative analyses to interpret and evaluate the node importance scores produced by GNNExplainer. Qualitatively, we visualize the node importance scores as heatmaps, applying colour grading to the meshes to emphasise the most significant regions contributing to predictions. Quantitatively, we generate box plots to compare node importance distributions across subjects and brain structures, both for raw scores and volume-adjusted scores. These analyses allow us to investigate label-specific trends, assess the impact of size bias on interpreting results, and provide a detailed understanding of GNNExplainer's outputs while highlighting potential limitations, such as its focus on node-level importance and susceptibility to size bias. In addition to analysing node importance, we also investigate shape variation to understand how the physical forms of these brain structures contribute to the model's predictions.

7.2.1 Model and datasets

The pre-trained model for sex classification was selected based on the model inspection study from Chapter 5. The model uses GCNConv, a non-shared submodel, applies rigid registration to the meshes and is trained on UKBB, where 15 subcortical brain structures[1] from T1-weighted brain MRI data are represented by meshes as an undirected graph composed of nodes, connected by edges forming triangular faces. We then use the external Cam-CAN [181, 199] dataset for testing. Please see Section 3.1.1 and 3.1.2 for datasets' details, respectively.

7.2.2 Substructure perturbation and model calibration

To complement GNNExplainer, we propose a perturbation-based method to evaluate the systemic impact of individual substructures on the model's decision-making. For each substructure, perturbation involves replacing its node features with the average

[1] Brain stem (BrStem), left/right thalamus (Thal), caudate (Caud), putamen (Puta), pallidum (Pall), hippocampus (Hipp), amygdala (Amyg), accumbens (Accu)

values while retaining the original features for other nodes. Key substructures are identified by analysing the most significant probability changes after perturbation. Additionally, graph-based models often produce over-confident predictions, even in uncertain cases, making explanations unreliable [75]. Hence, we apply post-hoc model calibration before reporting class-wise probability changes, considering only cases where the calibrated model made correct predictions, ensuring the reliability of the perturbation results. We compare calibrating the model using temperature scaling, Dirichlet calibration [111] and isotonic regression [167]. This approach provides a complementary perspective to GNNExplainer by focusing on the systemic impact of entire substructures rather than individual nodes or edges. It also addresses the issue of overconfidence in predictions, ensuring that the explanations are both interpretable and reliable.

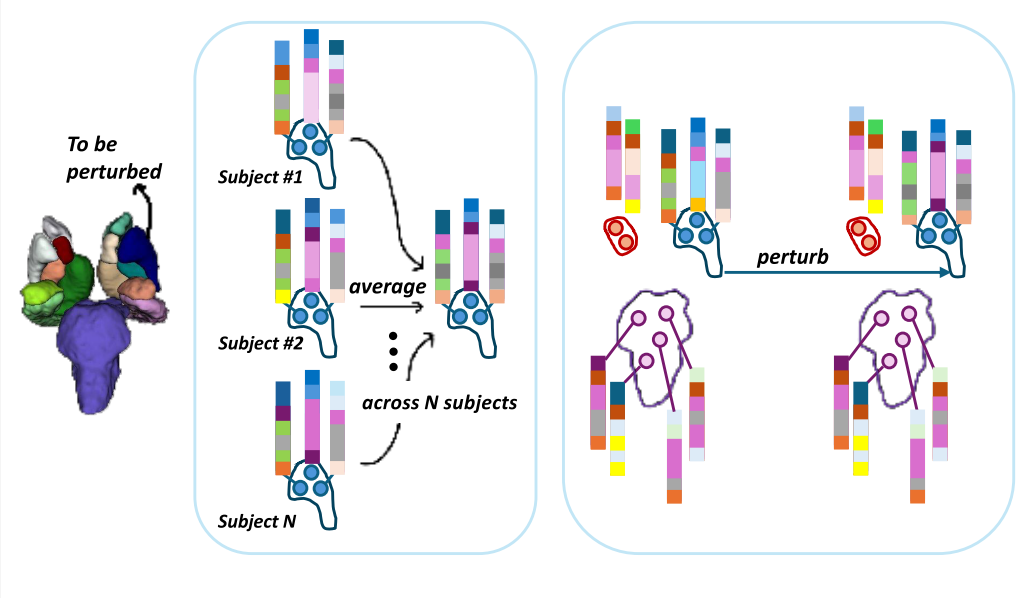


Figure 7.1: Perturbing the R_Puta The node features of all nodes in R_Puta are replaced by their average across the whole dataset. Node features from other substructures remain the same.

7.2.3 Complementary evaluations of importance scores

We compute node importance scores using GNNExplainer for both model and phenomenon configurations. We examine the importance scores within brain structures and across subjects to analyse how node importance varies, as described below. Results are visualised using box plots (see Fig. 7.2), grouped by biological sex to identify label-specific trends.

Node importance distribution (per substructure) For each subject, calculate the top-k importance values for each brain substructure, where k is a percentage of the total nodes within a substructure. The top-k values for each substructure are aver-

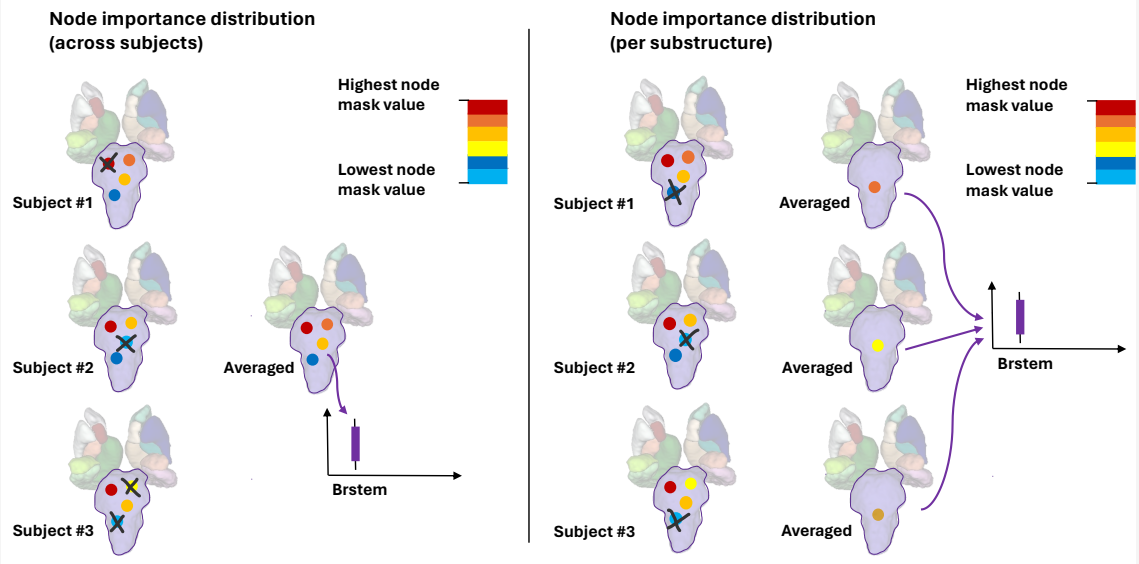


Figure 7.2: Generating the brainstem box plot using both node importance distribution methods by averaging across subjects (left) and per substructure (right). Crossed-out node mask values are not part of the top-k values.

aged, yielding a single score for that substructure to identify key substructures that consistently exhibit higher importance. These scores are normalised by dividing by the total sum of scores across all 15 brain structures for that subject. This step ensures that the 15 scores sum to one for each subject, facilitating the comparability of importance scores across all subjects, mitigating the effects of subject-specific variability, and identifying key substructures that consistently exhibit higher importance.

This method generates a vector of 15 scores for each subject (one score per brain substructure), where each value corresponds to the relative importance of one brain substructure. These vectors are then used to create box plots, which help identify which brain structures show consistently higher importance values across subjects.

Node importance distribution (across subjects) For each node per brain substructure, the top-k node mask values across subjects are selected, where k is a fixed percentage of 20%, 40%, 60%, 80% or 100% where 100% means using the node mask value from all subjects. These values are averaged across k% subjects to capture the general importance of nodes in that substructure. The k% subjects will differ from one node to another, Fig. 7.2(left). This method generates a vector for each brain substructure of a length equal to the number of nodes in that substructure, representing the relative importance of nodes. These vectors are used to create the target-label specific box plots, describing the node importance distribution per brain substructure. These plots help identify key structures where nodes consistently exhibit higher importance, offering insights into which areas of the brain are most relevant to the prediction

task.

Visualizing node importance in 3D - node masks The outputs are stored in the VTK format and rendered in ParaView. The visualisation highlights critical regions in each brain substructure, providing insights into how specific regions contribute to the model's predictions. Regions that consistently show high importance scores in the context of gender-based variations might correspond to brain areas strongly associated with the target label in the case of phenomenon explainability, or could indicate areas where the model's learned patterns are concentrated in the case of model explainability.

Volume normalisation For both methods mentioned above, instead of using the absolute raw node masks, we follow the same steps using the node mask values normalised to the volume of the brain substructure to which they belong. The volumes are available as part of the metadata that comes with the used datasets, per substructure, per subject. This method accounts for variations in substructure size, preventing larger substructures from dominating the importance scores. This allows for identifying truly influential substructures, irrespective of their size.

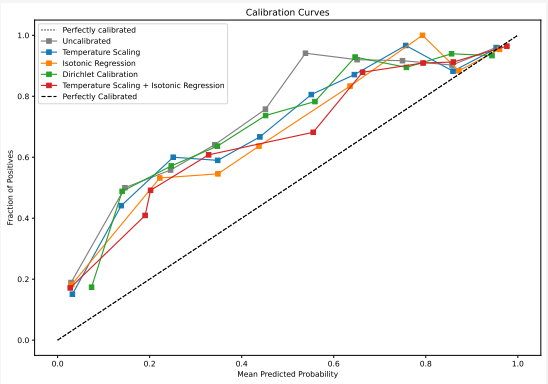
7.2.4 Shape variation

We applied PCA on each subcortical shape separately, allowing us to identify the most prominent modes of shape variation across the test set. The number of nodes per mesh times three for a single shape makes up the feature vector per subject $F_{subject}$. Therefore, the number of rows of the input data would be the number of meshes for that shape in the test set, and the number of columns would be $F_{subject}$. Singular Value Decomposition (SVD) was used to project this input matrix to a lower-dimensional space to identify the most prominent modes of shape variation.

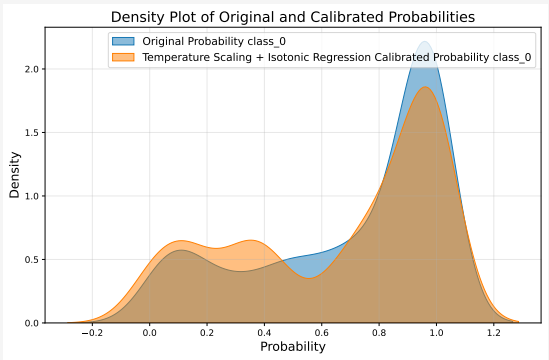
7.3 Results

7.3.1 Model calibration

To ensure a reliable classification threshold at 0.5, we apply post-hoc model calibration methods to our model predictions prior to pursuing our perturbation-based analysis. We selected the combined isotropic regression and temperature scaling as they are the closest to a perfectly calibrated model, as shown in Fig. 7.3a. We can also see that the threshold shifted towards 0.5 after calibration with a more defined dip in the distribution, Fig. 7.3b. After calibration, the number of correctly classified males increased from 162 to 204 after model calibration and decreased from 318 to 300 for females. As we are only interested in understanding how the model makes a correct prediction, we restrict both our perturbation-based analysis and the quantitative methods for analysing GNNExplainer node masks to the subset of correctly classified samples.



(a) Calibration curve comparing model calibration methods



(b) Density plot before and after model calibration

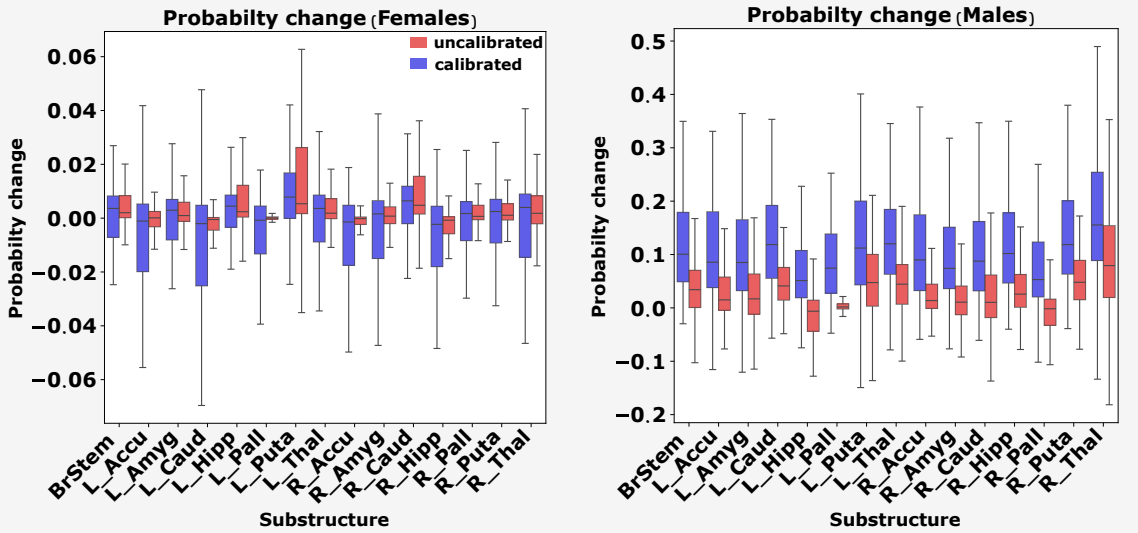
Figure 7.3: Model Calibration

7.3.2 Ranking using substructure perturbation

Fig. 7.4a, shows the average change in probabilities for predicting females correctly. We see that for predicting females L_Puta and R_Caud make the most significant change in the model prediction under perturbation. In Fig. 7.4b, we see that R_Thal makes the most significant change in predictions followed by R_Puta for the correctly classified males subset.

The magnitude of change in probabilities for females is very low compared to those for predicting males. This indicates that the model's predictions for females are less sensitive to perturbations than for males. After model calibration, we find that the model

for females is more ”confident” or robust in its prediction and, hence, less sensitive to perturbations. This could suggest that the model relies on different patterns in the data to perform the sex classification task, without necessarily associating specific features uniquely with either class.



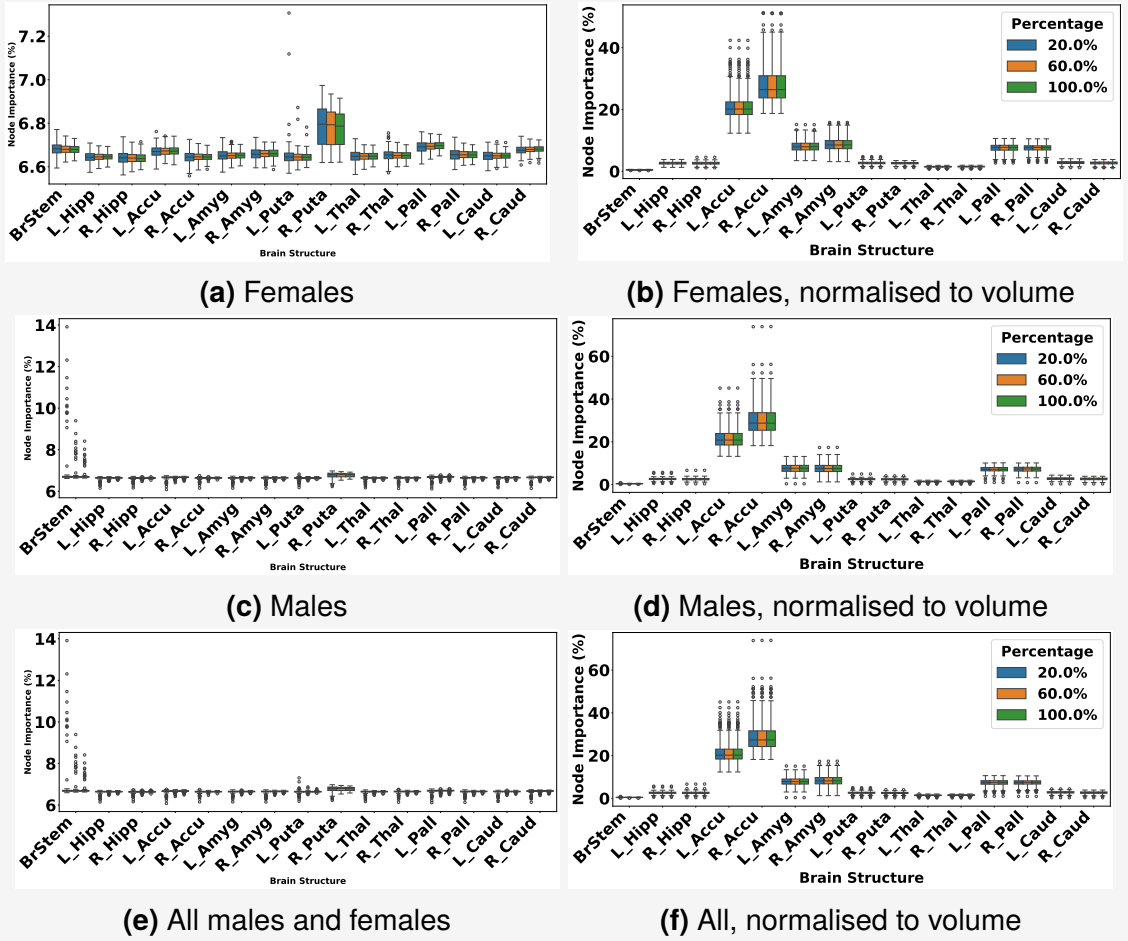
(a) Correctly predicted females **(b)** Correctly predicted males

Figure 7.4: Difference in probabilities before and after perturbation.

7.3.3 Analysing GNNExplainer importance scores

Node Importance Distribution (Per Structure) Fig 7.5a shows that R_Puta makes the highest contribution towards the decision-making process of the model to predict females by at least 0.1% more than other structures. We also see a higher variability in R_Puta's node importance scores for predicting both males and females Fig. 7.5c and 7.5a. As for predicting males, Fig. 7.5c shows that Brstem and R_Puta are the highest contributors.

Node Importance Distribution (Across Subjects) One interesting observation with this type of averaging is that we can see more significant gaps between the averages when using different k percentages and relatively small variance, most evident in Fig. 7.7a. For the same node, the average importance score decreases as the number of subjects included in the average increases, indicating that high scores (when only considering the top 20%) are likely due to a few extreme values. This pattern holds consistently across all structures. For predicting females, Fig. 7.7a shows R_Puta as the most important, followed by BrStem. Whereas Fig. 7.7c shows R_Puta and BrStem as most important for predicting males. Again, we also see a higher variability in R_Puta's node importance scores for predicting both males and females.



(a) Females **(b)** Females, normalised to volume

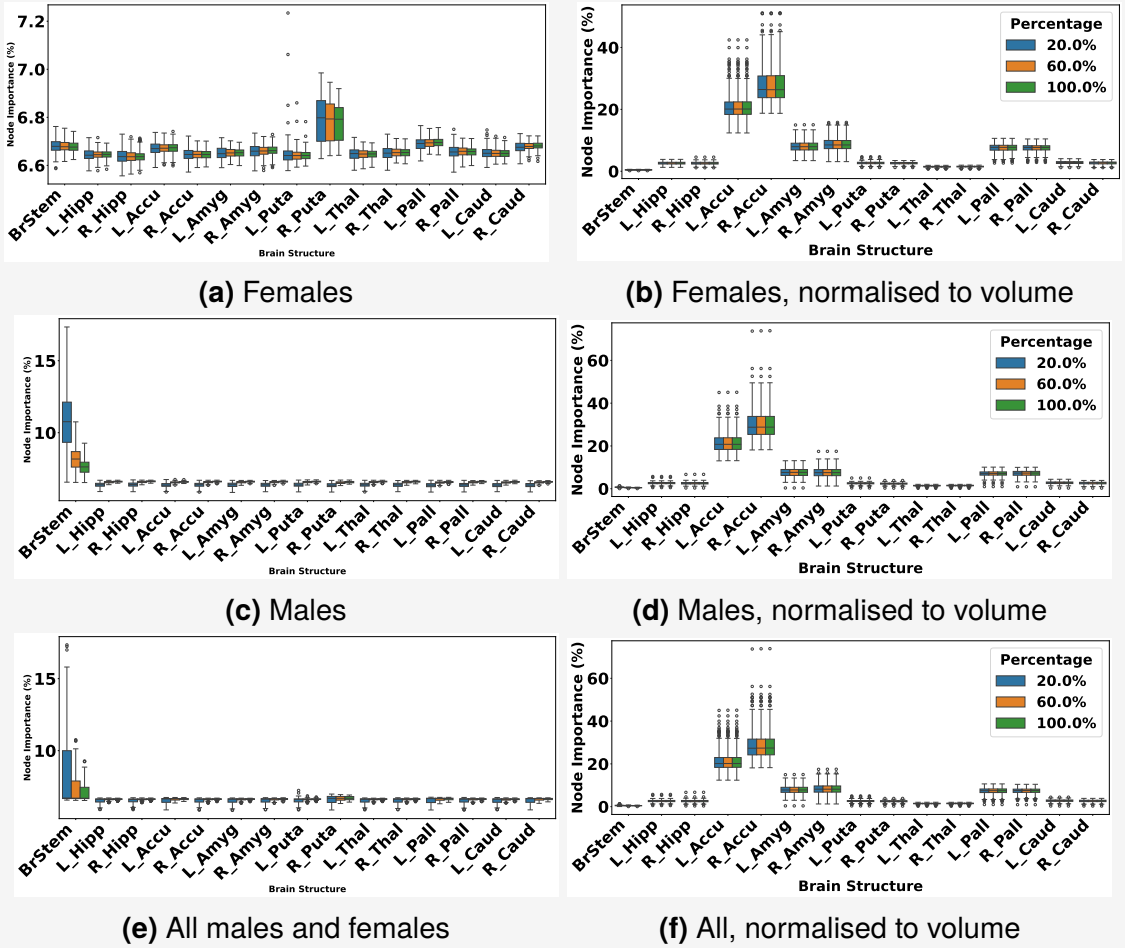
(c) Males **(d)** Males, normalised to volume

(e) All males and females **(f)** All, normalised to volume

Figure 7.5: Model Explainability (per substructure) top-k importance values for all nodes for sex classification. Left: Absolute importance values, showing label-specific variations. Right: Volume-normalized importance values, revealing consistent patterns across sexes, with L_Accu and R_Accu as key contributors. Volume normalization mitigates size bias, highlighting substructures critical for classification regardless of label.

Model vs Phenomenon explanations Both plots showed the same trends and magnitudes with and without volume normalisation. This similarity between the model explanations Fig. 7.5 and Fig. 7.7 and the phenomenon explanations Fig. 7.6 and 7.8 suggests that the importance patterns captured by the model are not arbitrary. Instead, they reflect intrinsic patterns in the data, indicating that the model is learning genuine relationships from the data, also evidenced by Fig. 7.9.

Effect of Volume normalisation Interestingly, when normalised to brain substructure volumes per subject, we find that for both male and female prediction, R_Accu and L_Accu are the most important contributors to the model's decision-making process, followed by R_Amyg and L_Amyg then R_Pall and L_Pall, shown in both averaging methods in Figs. 7.5b, 7.5d, 7.7b and 7.7d. This suggests that these substructures



(a) Females **(b)** Females, normalised to volume

(c) Males **(d)** Males, normalised to volume

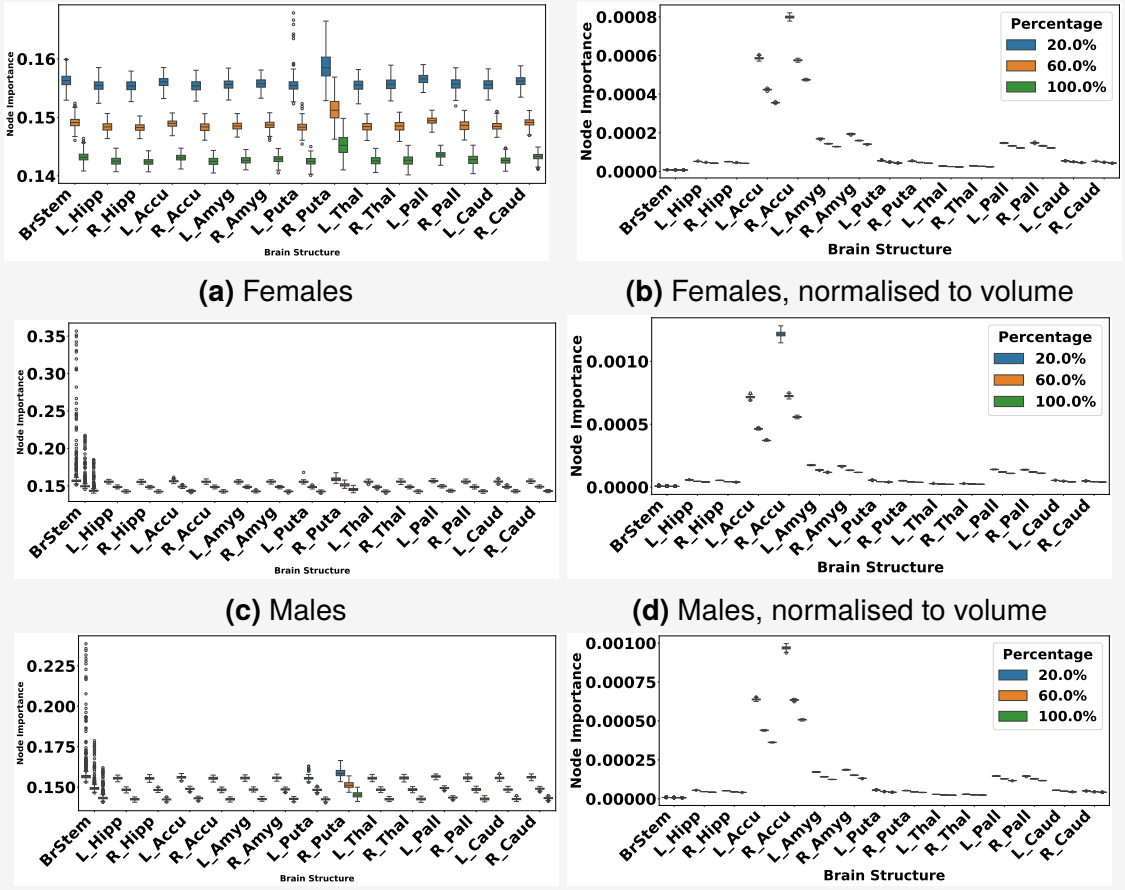
(e) All males and females **(f)** All, normalised to volume

Figure 7.6: Phenomenon Explainability (per substructure) top-k importance values for all nodes across subjects for sex classification. left: absolute averaged importance values, right: normalised to volume per brain substructure per subject then averaged

play a crucial role in classification regardless of the target label, further highlighting the benefits of volume normalisation in revealing consistent and generalisable patterns of node importance. This means that a bigger portion of the nodes making up these structures is important compared to structures bigger in size but having only a few important nodes, such as the Brstem, which can also be seen in Fig. 7.9.

7.3.4 Label-specific shape variation

For males, we see a wider range in variability across all top three PCA modes compared to females 7.10. The first mode captures high variability in the top part of the BrStem anterior and the bottom part of the BrStem posterior, as well as both tips of the L_Hipp and R_Hipp and the anterior tip of the L_Caud and R_Caud. The second mode captures a smaller change only on the sides of the brainstem and less on the sides of the L_Amyg and R_Amyg. The third mode shows variability across the entire



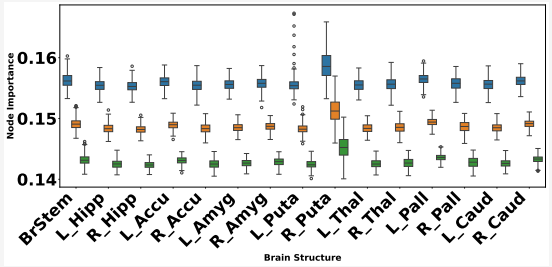
(a) Females **(b)** Females, normalised to volume

(c) Males **(d)** Males, normalised to volume

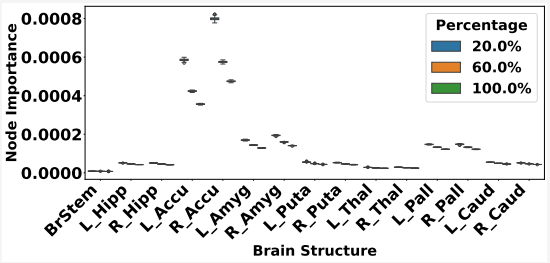
(e) All males and females **(f)** All, normalised to volume

Figure 7.7: Model Explainability (across subjects) top-k importance values for all nodes across subjects for sex classification. left: absolute averaged importance values, right: normalised to volume per brain substructure per subject then averaged

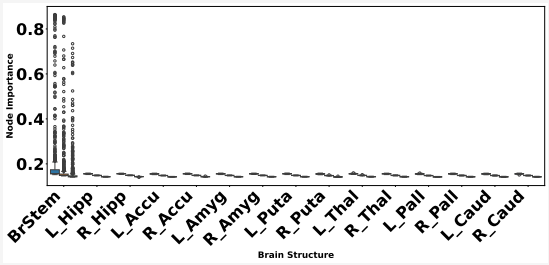
posterior surface and the top surface of the BrStem. The main differences between the subcortical brain shapes in males and females are the top part of the BrStem anterior, the sides of the BrStem, and the anterior tip of the L_Caud and R_Caud.



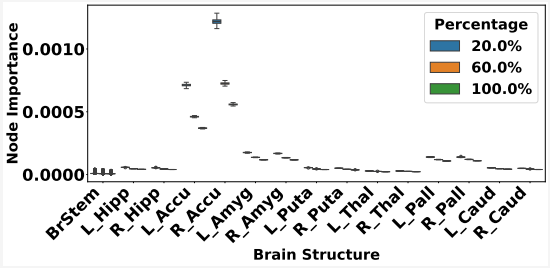
(a) Females



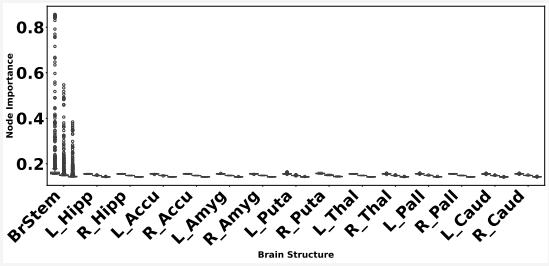
(b) Females, normalised to volume



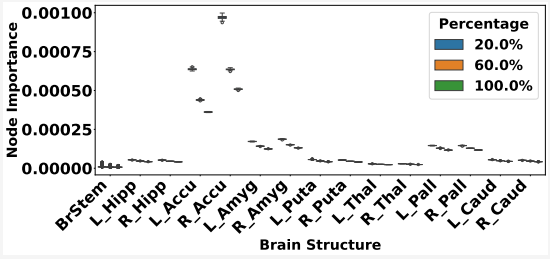
(c) Males



(d) Males, normalised to volume

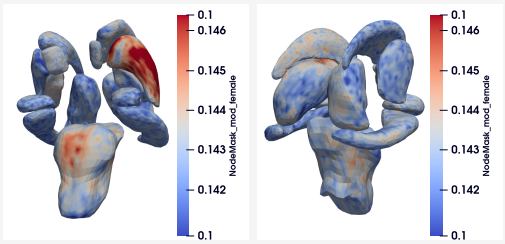


(e) All males and females

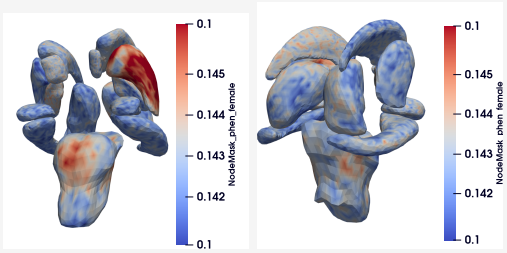


(f) All, normalised to volume

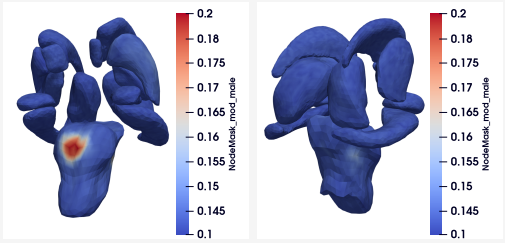
Figure 7.8: Phenomenon Explainability (across subjects) top-k importance values for all nodes across subjects for sex classification. left: absolute averaged importance values, right: normalised to volume per brain substructure per subject then averaged



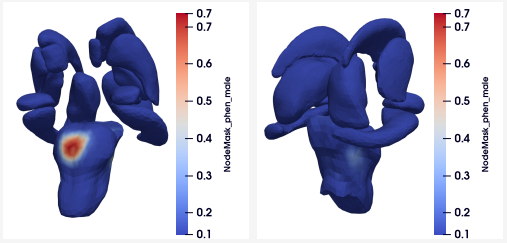
(a) Model explainability mask - females



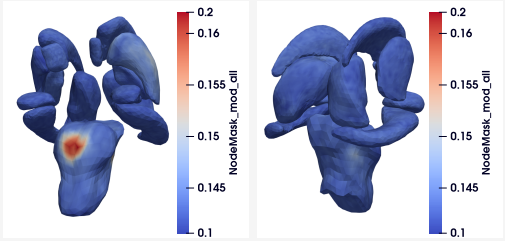
(b) Phen. explainability mask - females



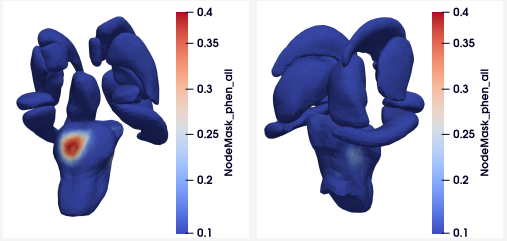
(c) Model explainability mask - males



(d) Phen. explainability mask - males

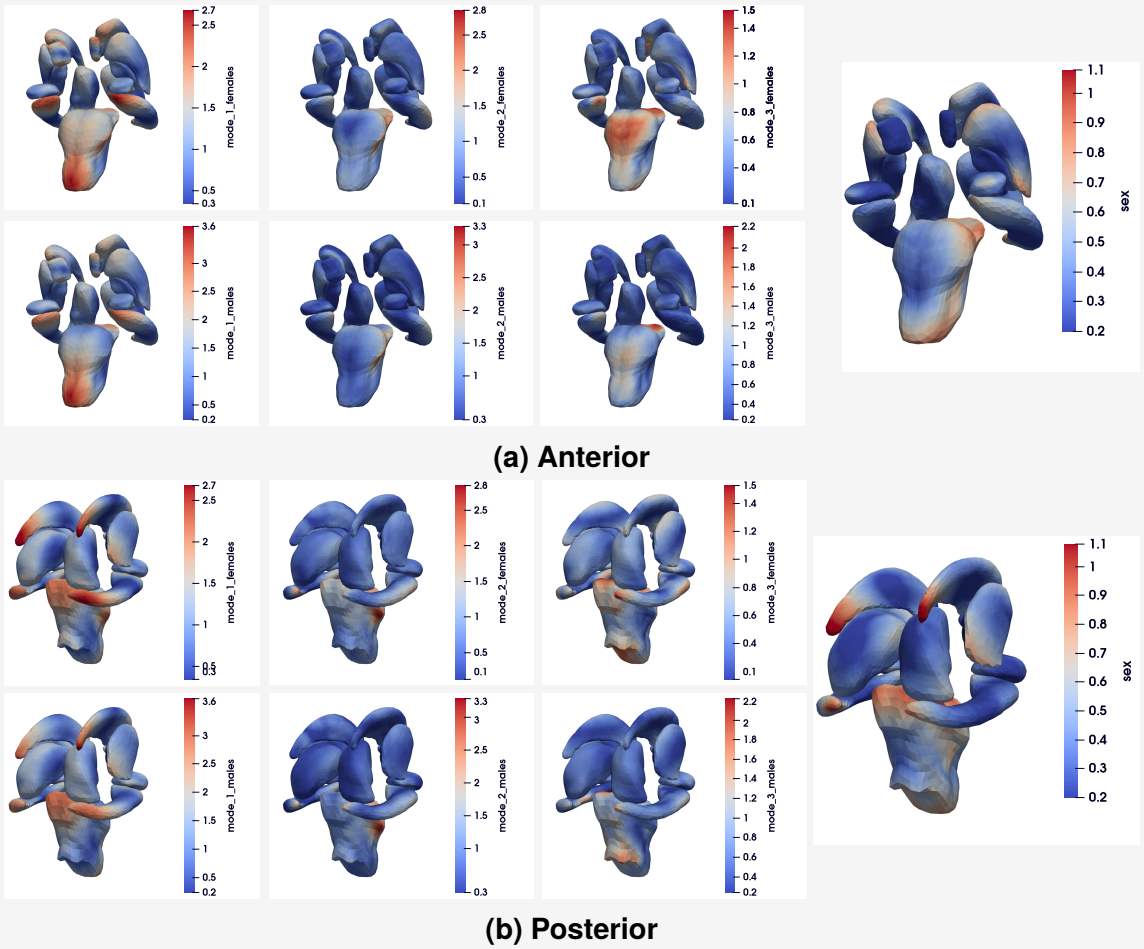


(e) Model explainability mask - males and females



(f) Phen. explainability mask - males and females.

Figure 7.9: The anterior and posterior views are shown on the left and right, respectively, for each configuration (model or phenomenon), explainability and each subgroup. The top row is for females (a) and (b)), the middle row is for males (c) and (d), and the bottom row is for both males and females (e) and (f). The importance scores are colour-coded (low blue, high red) and superimposed on the average shapes.



(a) Anterior

(b) Posterior

Figure 7.10: Top PCA modes and the difference between mean males and females. Please note that the colour bars have different ranges so that the variability is visible.

7.4 Conclusion

This study addresses the limitations of current GNN explainability methods by proposing a calibration-aware, multi-level framework that addresses key limitations of existing methods by aligning explanations with model confidence. Our approach investigates model-level and phenomenon-level explanations aggregated in two different ways, as well as our custom perturbation-based approach to identify critical brain substructures for sex classification.

Our findings reveal significant discrepancies between methods, with GNNExplainer emphasising BrStem and R_Puta for males and R_Puta for females, while the perturbation-based method highlighted R_Thal for males and L_Puta and R_Caud for females. The perturbation-based method also showed that the model is less sensitive to perturbations when predicting females than for males, indicating the model is more confident with female predictions.

Volume normalisation, employed to mitigate size biases across substructures, further uncovers consistent importance patterns across labels, suggesting inherent biases in size-driven explanations. Otherwise, the results reveal label-specific patterns. Sex-based shape differences in BrStem shown by PCA may explain its outsized role in male predictions.

These learning outcomes arise because the model captures consistent, discriminative morphological variations in these substructures that align with sex-specific patterns in the data. However, the differences between methods occur because each method quantifies 'importance' differently: GNNExplainer prioritises node-level feature sensitivity and is influenced by local gradients and structural size, while the perturbation-based method evaluates the systemic impact of removing or altering entire substructures on predictions, which can highlight different regions critical for maintaining the model's confidence. Additionally, volume-adjusted analyses showed that some initial high importance scores (e.g., R_Puta) may partially reflect size bias rather than true discriminative power, leading to different rankings once adjusted. These methodological differences underscore that while all methods capture meaningful biological signals, they illuminate different facets of what the model has learned, explaining why the stipulated outcomes differ across methods while still aligning with known sex-related shape patterns [173].

Future studies should consider integrating and cross-validating findings across multiple approaches to better understand the role of specific brain substructures in sex classification. Also, confirm findings for other tasks.

Chapter 8

Automated Segmentation of the Aorta - E-HVols

To extend the application of our methods developed for shape analysis from Chapter 4 to cardiac imaging, we investigate the use of a raw dataset of MR images collected as part of the E-HVols study. In this chapter, we address several challenges, including the automation of aortic segmentation from MR images and the difficulties associated with using real-world datasets, which often lack the standardisation and controlled conditions found in academic or publicly available datasets. To overcome these challenges, we developed a custom data pre-processing pipeline that emphasises the critical importance of maintaining consistent image orientation across the entire dataset.

Aortic shape, size, and kinematics are critical to its function. Characterising the aorta's phenotype and its determinants is essential for understanding the interrelationships between shape, size, and function. Recent advances in medical imaging have enabled the extraction of detailed morphological and functional data, providing a foundation for automated analysis pipelines. These pipelines integrate state-of-the-art segmentation models, mesh processing techniques, and statistical shape analysis to capture the complexity of the cardiovascular system.

For the remainder of the thesis, we aim to address the following three objectives:

1. Automate the aortic segmentation and 3D surface extraction from cardiac MRI.
2. Establish a robust pipeline for aortic mesh processing, including registration and statistical shape analysis, to enable precise morphological studies.
3. Collect complementary data such as demographics and function-related parameters to establish a benchmark for aortic phenotypes.

Plan of Investigation:

1. Evaluate state-of-the-art segmentation models on the E-HVols dataset.
2. Compare traditional and deep learning-based registration techniques (e.g., ICP, RANSAC, CPD, and proposed deep learning methods) using metrics like RMSE, Hausdorff distance, and Dice coefficient to ensure accurate vertex correspondence and mesh quality.
3. Investigate the relationship between aortic shape, size, and function by applying Principal Component Analysis (PCA) to quantify aortic shape variability and investigate correlations between PCA modes and demographics, aortic size-related parameters and left ventricular functional variables using linear regression and our GNN models.

This chapter and the following one are based on the paper accepted for STACOM 2025: Shehata N, Elsawy A, Nagy M, ElMahdy M, Romeih S, Aguib H, Yacoub M, Glocker B. "A Comprehensive Pipeline for Aortic Segmentation and Shape Analysis".

8.1 Introduction

Most research conducted on 3D aortic segmentation or shape analysis has utilised images from CT [136, 179], with only a few studies using MRI [83] for these purposes. Analysing MR images is more challenging compared to high-resolution CT due to several factors, including the similarity in intensity and weak edge differentiation between the aorta and the surrounding structures [8], as well as varying image quality and inconsistent image contrast [82]. This is evidenced by the reported dice scores from the Multi-Modality Whole Heart Segmentation (MM-WHS) Challenge, where higher dice scores were achieved on CT images compared to those from MR [250].

In this chapter, we evaluate five state-of-the-art models to automate the segmentation of the thoracic aorta from MRI collected under the E-HVols study (see Section 3.1.5). We compare nn-UNet [92], TotalSegmentator [37], MedSAM2 [134], and two variations of the fine-tuned TotalSegmentator (`FT_TS` with frozen encoder layers and `FT_TS_NOFREEZING` with no frozen layers).

8.2 Automated segmentation

In this section, we describe the process of automating the segmentation of the aorta from CMR images using a combination of semi-automated tools and deep learning models. The aim is to develop an efficient and reproducible pipeline for extracting accurate aortic segmentations, which can then be used for further analysis.

Ground-truth Segmentation for Training CMR images were imported into Mimics Research version 19.0 (Materialise, Leuven, Belgium) for semi-automated aorta segmentation and 3D reconstruction. Labelling was performed on 200 out of 744 image sets due to the growing dataset size and the time-consuming nature of the task. To ensure consistency across all samples, ground-truth annotations were limited to the descending aorta down to the diaphragm level.

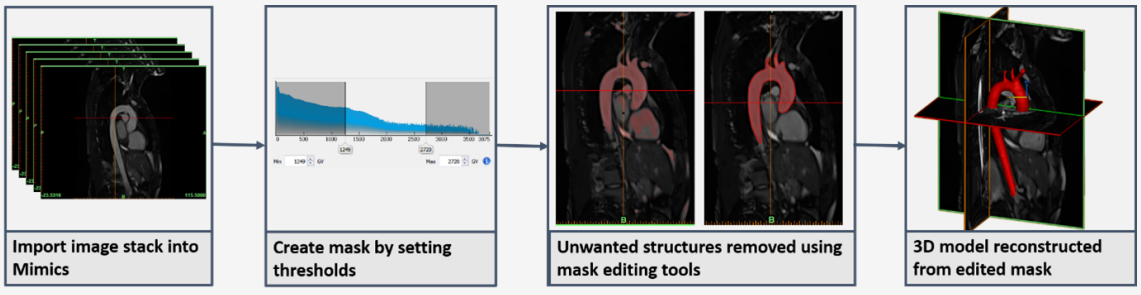


Figure 8.1: Using Mimics for 3D reconstruction of the aorta.

DICOM to NIfTI conversion The conversion was performed using `MRIcroGL` which offers a GUI based on `dcm2nii` [1]. This was necessary to be able to use some of the state-of-the-art models listed in section 2.3. NIfTI also anonymises the subject's information and takes up less disk space than raw DICOM data.

Exporting Masks from Mimics This step was carried out in two phases. Initially, the segmentation results from Mimics were saved as STL files, which did not directly provide segmentation masks. As a result, we needed to extract the segmentation masks from these STL files. This process involved the following steps:

1. **Mask Generation from STL Files:** The first step was to calculate the mask from the STL file and overlay it onto the original DICOM images. Afterwards, the modified DICOM images (with the mask overlaid) were saved.

2. **Binary Mask Creation:** In the second step, we created a binary mask by subtracting the newly saved DICOM images (which had the mask overlaid) from the original DICOM images (without the mask). This subtraction process yielded a

[1] https://www.nitrc.org/projects/mricrogl

binary mask, which, along with the relevant headers, was then saved in NIfTI format using the Python package `nibabel`.

8.2.1 Image pre-processing

The images were reoriented consistently into a standard Left-Posterior-Superior (LPS) coordinate system for fine-tuning MedSAM2 and Right-Anterior-Posterior (RAS) for nn-UNet, as inconsistent orientation can cause the models to perform poorly. This was added to the pre-processing code from MedSAM2, but for nn-UNet, this was achieved by setting the reader to *NibabelIOWithReorient* in the `dataset.json` file, which contains the metadata nn-UNet needed for training. Additionally, intensity normalisation was applied by clipping outliers to their 0.5 and 99.5 percentiles and rescaling the intensity values to a standard range of 0-255 while preserving zero values.

8.2.2 Model selection for segmentation inference

We evaluated five models for segmenting the aorta from MRI (please refer to section 2.3 for more information on these models): nn-UNet [92], TotalSegmentator v2.4.0 [37], MedSAM2 [134], and two fine-tuned versions of TotalSegmentator (`FT_TS` and `FT_TS_NOFREEZING`). `FT_TS` involved freezing the early encoder layers during fine-tuning, leveraging pre-trained feature extraction while updating the latter layers only. In contrast, `FT_TS_NOFREEZING` allowed all layers to be updated during training, enabling complete adaptation to the dataset. We trained the models using 5-fold cross-validation on the 200 annotated scans. The best-performing model is then used for large-scale inference and quality-controlled manual inspection. The ensemble models were evaluated based on Dice score, precision, and recall.

What is fine-tuning a model? Transfer learning is a technique where a model trained on one task is reused as a starting point for a model on a second, related task. Freezing early layers is a common technique in transfer learning. It assumes that these layers have learned general, low-level, transferable features across tasks. This involves:

- Pre-training: A model is first trained on a large dataset for a general task.
- Knowledge transfer: The pre-trained model, with its learned features and patterns, is then applied to a new, often more specific task.
- Fine-tuning: The model is further trained on the new task's dataset, adapting its knowledge to the specific requirements.

This approach is beneficial when the target task comes with limited data, as it leverages knowledge gained from a broader dataset. Only part of the model (usually the later layers) may be retrained, while earlier layers retain their original learned features.

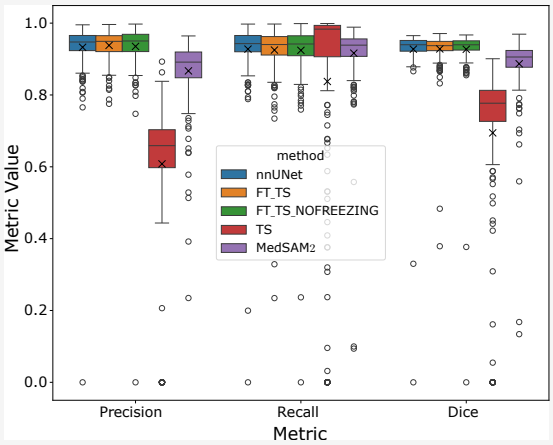
8.2.3 Quality control

The best-performing model was used to segment the remaining 544 images, which were manually inspected as they did not have ground-truth segmentation. The image mask pairs were superimposed and saved as GIFs to be displayed for inspection.

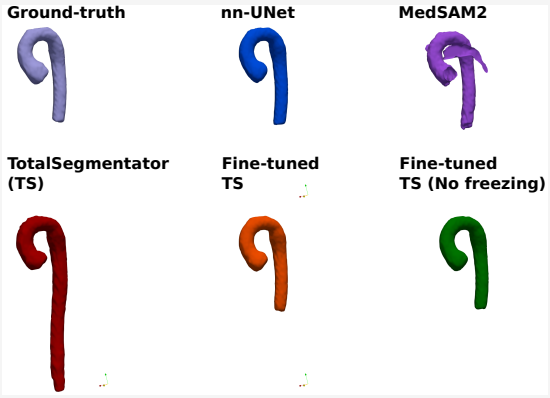
8.3 Results

Fig. 8.2a shows a box plot comparing the performance of the five ensemble models using precision, recall and dice score.

An example of the segmentation masks produced by each of the five models for a random sample in the dataset is shown in Fig. 8.2b.



(a) Segmentation performance.



(b) Segmented random sample.

Figure 8.2: Segmentation results.

8.3.1 TotalSegmentator

High recall with lower precision typically indicates oversegmentation. This means the method identifies most of the target region (high recall) and includes extra areas that do not belong to the target region (lower precision). The non-fine-tuned TotalSegmentator shows this due to the difference in ground-truth segmentation. See Fig. 8.2b top row (Ground-truth vs TotalSegmentator).

8.3.2 MedSAM2

Similarly, fine-tuned MedmSAM2, although prompt-dependent, seems to be slightly oversegmenting, but generally performs better than TotalSegmentator in terms of precision and dice. Please refer to section 2.3 for more details on these models.

8.3.3 Fine-tuned TotalSegmentator

We compare fine-tuning with freezing half of the encoder layers and without freezing any layers, referred to as `FT_TS` and `FT_TS_NOFREEZING`, respectively, in Fig. 8.2a. We find that both models show very similar performance across all metrics. This suggests

that the freezing part of the encoder does not significantly impact the model's performance in this case, and that the unfrozen layers (the second half of the encoder layers and the entire decoder) can still adapt to the specific segmentation task.

8.3.4 nn-UNet

nn-UNet was trained solely on our dataset, the E-HVols, without any pre-training. Unlike TotalSegmentator, which shares a similar architecture but was pre-trained and then fine-tuned, nn-UNet was optimised from scratch, likely accounting for its superior performance. This model showed slightly better precision, recall, and Dice scores compared to both fine-tuned TotalSegmentator versions (`FT_TS` and `FT_TS_NOFREEZING`), indicating that full-layer training on the task-specific dataset yields marginal improvements despite increased computational costs. It also yields only two outliers with dice scores below 0.75.

8.3.5 Failed segmentation by nn-UNet - test data

Of the 544 test images, 18 were incorrectly segmented by nn-UNet, due to acquisition problems, reducing our total dataset size from 744 to 726. The missing parts of the segmentation are where the crosshairs are placed, Fig. 8.3. Most of the incorrect segmentations were due to signal loss in the ascending aorta or aortic arch during image acquisition, or where the images did not capture the aorta fully (e.g. case #3). Whereas cases (#7, #8, #9 and #10) seem to be hazy.

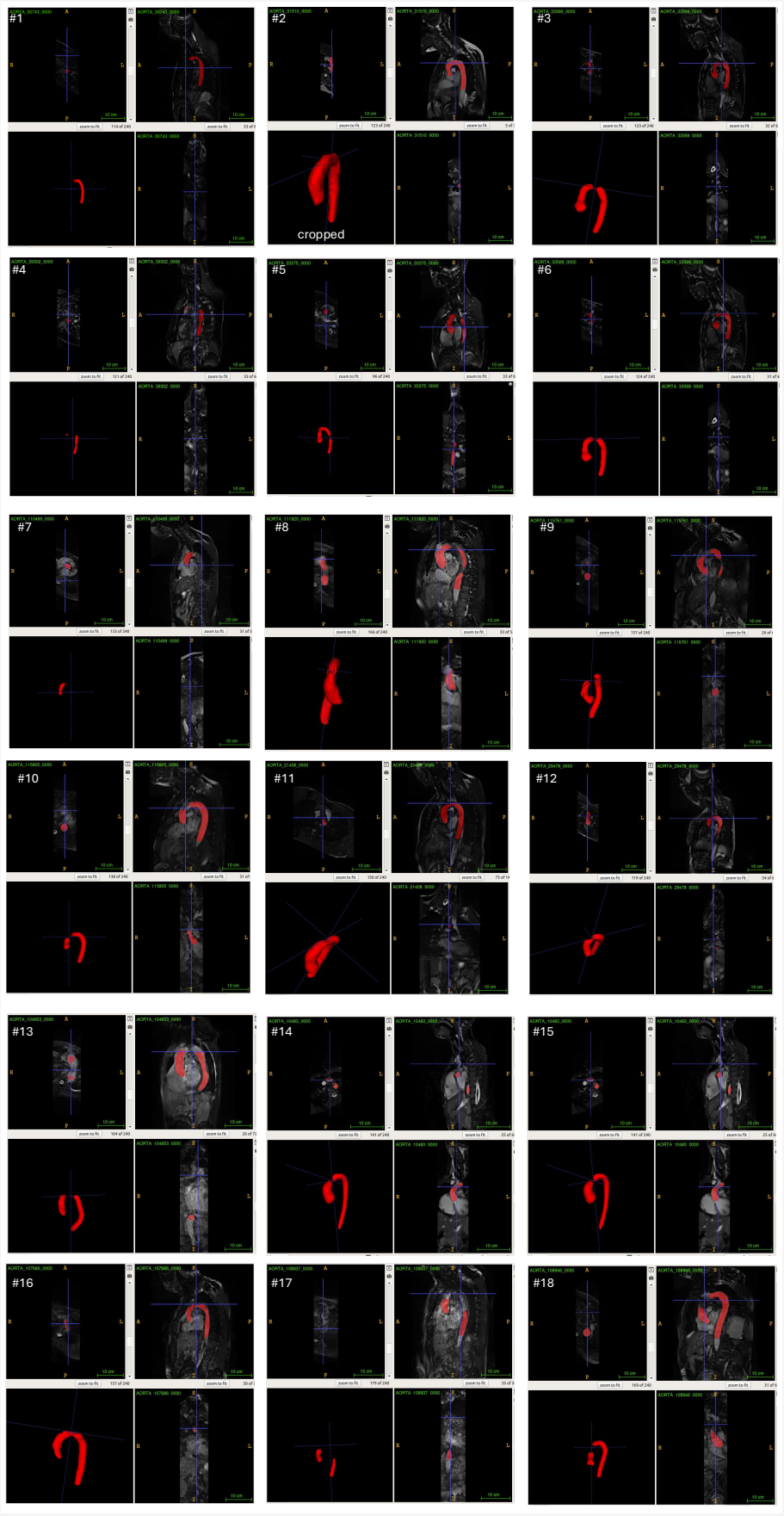


Figure 8.3: Failed segmentation using nn-UNet for inference.

8.4 Conclusion

The E-HVols dataset includes consistent ground-truth annotations, limiting the descending aorta to the diaphragm level, which influences 'pre-trained' model performance, as was demonstrated by TotalSegmentator. The ground-truth annotations are different from those used for pre-training. Therefore, TotalSegmentator benefited a lot from fine-tuning.

The similar performance between `FT_TS` and `FT_TS_NOFREEZING` indicates that freezing portions of the model allows efficient transfer learning while preserving generalisation, maintaining a good balance between leveraging pre-trained features and adapting to the new task.

Fine-tuned MedSAM2, although prompt-dependent, performed comparably to fine-tuned TotalSegmentator, with slight degradation in precision and dice, and showing tendencies for oversegmentation. MedSAM2 may benefit from a postprocessing module utilising connected-component labelling to discard the extra detached blobs from the aorta. However, this has not been pursued as MedSAM2 can not be used for inference, as the ground truth is not available to be used for prompting.

Full training, as with nn-UNet, can offer slight performance gains over fine-tuned TotalSegmentator, but it also comes with higher computational requirements and a potential risk of overfitting. However, for our use case, we chose nn-UNet for its highest average metric scores, despite the complexity of additional training.

This analysis provides insights into the effectiveness of transfer learning and layer freezing in medical image segmentation tasks, showing that strategic use of pre-trained features and task-specific tuning can yield robust results while balancing computational resources.

Chapter 9

Aortic Mesh Processing and Atlas Creation

This study presents a comprehensive pipeline for processing and analysing aortic meshes extracted from medical imaging segmentation. The workflow encompasses iso-surface extraction using the Marching Cubes algorithm and mesh post-processing steps. Then, finally, the critical step of rigid/non-rigid registration is used to establish vertex correspondence across the extracted meshes in the dataset. We compare the performance of various methods, including ICP (Point-to-Point and Point-to-Plane), RANSAC with FPFH features, CPD affine and deformable registration, and a proposed deep learning approach that employs a composite loss function.

We evaluate all registration methods using RMSE, Hausdorff distance, IoU, Dice coefficient, and average point-to-point distance. All metrics favoured the deep learning approach, demonstrating superior performance in establishing accurate correspondence while preserving mesh topology and geometric features. Additionally, the surface extraction and mesh processing steps in the presented pipeline ensure high-quality meshes from segmented images, promoting the use of GNNs for downstream tasks. This comprehensive framework provides a robust foundation for subsequent statistical shape analysis and morphological studies of the aorta, with potential applications in understanding anatomical variations and pathological conditions. We demonstrate this application by studying shape variations using PCA.

9.1 Introduction

The reconstruction and processing of 3D surfaces from medical imaging data is a critical task with applications spanning medicine, robotics, and computer vision. Accurate 3D surface representation demands highly resolved surfaces that are smooth, watertight, robust to noise, and capable of capturing fine details. Classical methods often face limitations in scalability and preserving fine details in large or noisy datasets [54]. Recent advances in deep learning have opened new avenues for addressing these challenges by enabling more robust and flexible 3D reconstruction and registration workflows.

The reconstruction of 3D surfaces from point clouds and other forms of input data has seen significant progress, leveraging both traditional geometric methods and modern deep learning-based approaches. Classical algorithms, such as Poisson Surface Reconstruction [102], the Ball-Pivoting Algorithm [13], Delaunay Triangulation [17], and Voronoi Diagrams [58], have been widely used for generating smooth and watertight 3D meshes. These methods rely on strong geometric priors and effectively create surfaces that align with the input data. Other techniques [4, 6, 51] have further advanced reconstruction by addressing the relationship between surface reconstruction and ϵ-sampling. However, many of these approaches struggle with scalability, handling noise, and undersampled regions, making them less suitable for reconstructing complex or irregular shapes. Extensions like COCONE [5] and TIGHTCOCONE [46] incorporate strategies to address surface boundaries and fill holes, but they often require additional post-processing.

What is epsilon sampling? A technique used to sample points from a 3D object (such as a surface) in a way that ensures all key features of the surface are captured accurately. For a cloud of points representing the surface of an object, ϵ refers to a small distance threshold that helps determine how far apart points should be for efficient sampling. The goal is to pick points spaced at least epsilon apart from each other. This helps avoid having too many points that are too close together while preserving important surface features. This also makes the reconstruction process of the 3D shape more efficient.

Deep learning has emerged as a powerful alternative, offering the ability to learn directly from data and overcome limitations inherent to classical methods. For instance, Groueix et al. [73] introduced a method for learning 3D surface generation by deforming a template mesh called the ”papier-mâché” approach. Liu et al. [127] proposed a framework for meshing point clouds using intrinsic-extrinsic (i.e. geodesic-

Euclidean) ratio guidance, demonstrating improved accuracy in reconstructing fine-grained geometries. Daroya et al. [40] developed a flexible mesh generation pipeline, emphasising adaptability to various input data types. Others incorporate meshlet priors to improve geometric consistency [10]. Deep learning adaptations of classical methods include Deep Marching Cubes for explicit surface learning [122] and Scan2Mesh for structured mesh generation from range scans [38]. Further, techniques like POINTS2SURF and POINTTRINET refine reconstructions using local and global structures [53, 183]. However, they often involve trade-offs between geometric and topological precision. While Point2Mesh leverages self-priors for surface refinement [81]. Mesh-based and graph-based approaches integrate deep learning for detection and segmentation, such as DMNET using GNNs for surface reconstruction [235] and Mesh R-CNN combining mesh generation with object detection to predict and reconstruct 3D shapes in an end-to-end manner simultaneously [67].

These advancements demonstrate the significant progress in 3D surface reconstruction, from classical geometric methods to modern deep learning-based approaches. However, challenges remain in ensuring scalability, robustness to noise, and the preservation of fine details, particularly in medical applications such as anatomical shape analysis, where precision and robustness are paramount. Please refer to Section 2.6 for background on registration.

The workflow, Fig. 9.2, involves iso-surface extraction via Marching Cubes [128], followed by mesh post-processing to ensure uniform point cloud densities and advanced registration techniques for vertex correspondence. We explore rigid and non-rigid registration using ICP (Point-to-Point [150], Point-to-Plane [30]), RANSAC with FPFH [246], CPD [142], and a proposed deep learning approach with PyTorch3D [164], enforcing topological consistency for optimised registration. A template mesh deforms to target meshes, with template selection minimising bias. Establishing correspondence (Section 2.6) enables shape variation analysis via PCA. Performance is assessed using RMSE, Hausdorff distance, IoU, Dice, and point-to-point distance (Section 2.7), confirming deep learning's superiority over classical methods. Our contributions enhance reconstruction and registration robustness, laying the groundwork for statistical shape analysis, as demonstrated by investigating aortic shape variation using PCA. The proposed surface reconstruction workflow facilitates mesh extraction from medical images, promoting GNN applications. This work tackles the challenge of accurate vertex correspondence while preserving geometric and topological integrity, highlighting the importance of leveraging traditional and data-driven approaches in advancing medical shape analysis and understanding anatomical variability and pathology.

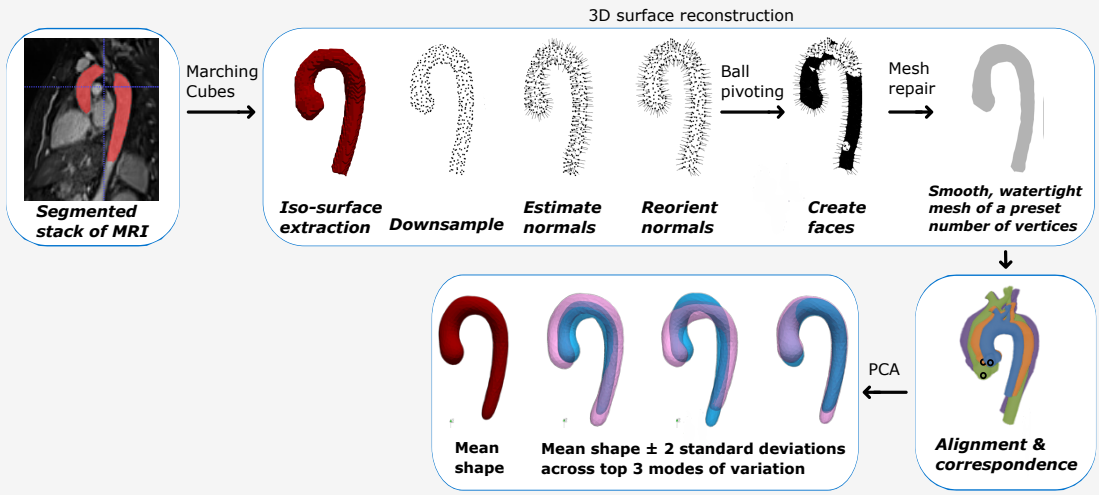


Figure 9.1: Overview for the developed pipeline starting with extracting high quality meshes from segmented images to applying mesh registration, followed by PCA for shape analysis.

9.2 3D surface reconstruction

After the aorta was successfully segmented, some additional steps were necessary to generate a mesh from these masks. Additionally, the meshes had to be downsampled, aligned and registered to apply PCA as will be described in the next section 9.4. These steps are summarised in Fig. 9.2.

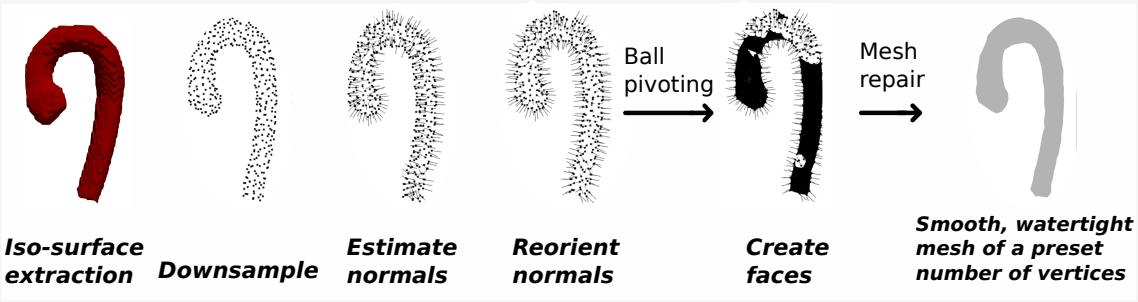


Figure 9.2: Overview for 3D surface reconstruction steps following segmentation. Generated using Open3D

9.2.1 Iso-surface Extraction

An accurate iso-surface is generated from the mask using the *Marching Cubes Algorithm* that converts voxel data into a mesh, taking into account problems like partial volume effect [64] where the object may be underestimated. The implementation by *scikit-image* was used, and the final result was a mesh defined by a list of vertices and faces for each mask object. A 3D mesh is represented as a collection of vertices and faces:

$$\mathcal{M} = (\mathcal{V}, \mathcal{F})$$

where:

- $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N\}$ is the set of vertices, $\mathcal{V} \in \mathbb{R}^{N\times 3}$.
- $\mathcal{F} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_M\}$ is the set of faces, each defined by a triplet of vertex indices (i, j, k), $\mathcal{F} \in \mathbb{N}^{M\times 3}$.

9.2.2 Post-processing

Each mesh generated by marching cubes was represented by a different number of vertices. The meshes had a fixed number of vertices across all subjects to perform PCA. Also, for computational overhead, downsampling was preferred. The following post-processing steps were applied to every mesh prior to alignment and registration (all methods).

- **Downsampling vertices** the vertices were sampled to be somewhat equidistant, the method is based on Poisson disk sampling [233] and the implementation by *open3d*[1] was used. The fixed number of vertices was set to 1000, but the method was not robust and sometimes yielded a few more or fewer vertices than 1000.
- **Estimate and re-orient normals** To recreate faces after downsampling, normals are estimated by fitting local planes to the 3D points [151] and reoriented using the tangent plane propagation method, using Open3D implementation based on the minimum spanning tree [247]. This step resolves ambiguities in local surface orientation, preventing issues such as inward-pointing normals or holes when reconstructing the mesh. However, a few normals may still remain pointing inwards due to ambiguities in the local surface orientation.
- **Reconstruct mesh from downsampled points** Ball pivoting algorithm [13] implemented in *open3d* was used to reconstruct the faces for the newly downsampled points, given coherent normals. Multiple radii are used during this process to accommodate varying point densities and ensure robust face reconstruction, *radii [0.05, 0.1, 0.2, 0.4, 0.8, 1.0 to 12.0]*.
- **Mesh repair** To ensure a high-quality manifold surface mesh, a mesh repair routine was called by *pymeshfix* [2]. This includes (1) filling holes by creating additional faces, ensuring the mesh is watertight and does not contain any open boundaries, (2) removing non-manifold edges, which occur when an edge is shared by more than two faces, (3) inverting faces with incorrect normals, (4) removing disconnected components and (5) removing faces with near zero areas when all three vertices are almost co-linear or too close to each other and (6) smoothing to improve mesh quality and remove any artefacts.
- **Reinstate vertices count** The target vertex count is set to 1000, although slight variations may occur due to the mesh repair steps and the stochastic nature of the sampling method. Therefore, vertices were added at the midpoint of the longest edges and were referenced by two new faces instead of the old one.

[1] https://www.open3d.org/
[2] https://pymeshfix.pyvista.org/

9.3 Alignment and Correspondence

The alignment of the meshes is only performed once, but the registration to establish correspondence between vertices of one mesh to another across the dataset was performed twice, to a subset (K meshes) of the whole dataset for template selection and another time for the whole dataset using the saved template from the first run; figure 9.3.

- **Alignment - Spatial normalisation** Volunteers were placed in the MRI machine the same way, allowing for approximate alignment of their aortas, with the z-axis oriented along the descending aorta. This consistent positioning meant that only translational adjustments were necessary to align the aortas precisely. To determine these translations, we calculated the centroid of the template mesh and then computed the displacement vector from this centroid to the centroid in each subject's aortic mesh.

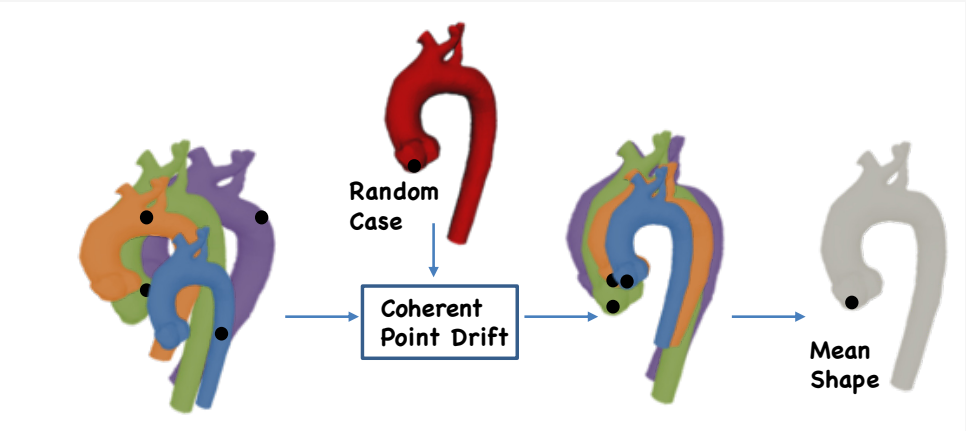
  Alignment involves translating the target mesh to align with the source mesh's centroid: $\mathbf{T} = \boldsymbol{\mu}_{\text{source}} - \boldsymbol{\mu}_{\text{target}}, \quad \mathcal{V}_{\text{target}}^{\text{align}} = \mathcal{V}_{\text{target}} + \mathbf{T}$

- **Template selection** We registered a subset of $K = 100$ meshes to an initial random mesh from the dataset using affine Coherent Point Drift (CPD). The vertices of the K registered meshes were averaged to create a mean shape, which served as the final template, see Fig. 9.3. This approach minimises bias toward any specific shape and provides a canonical representation for downstream analysis.

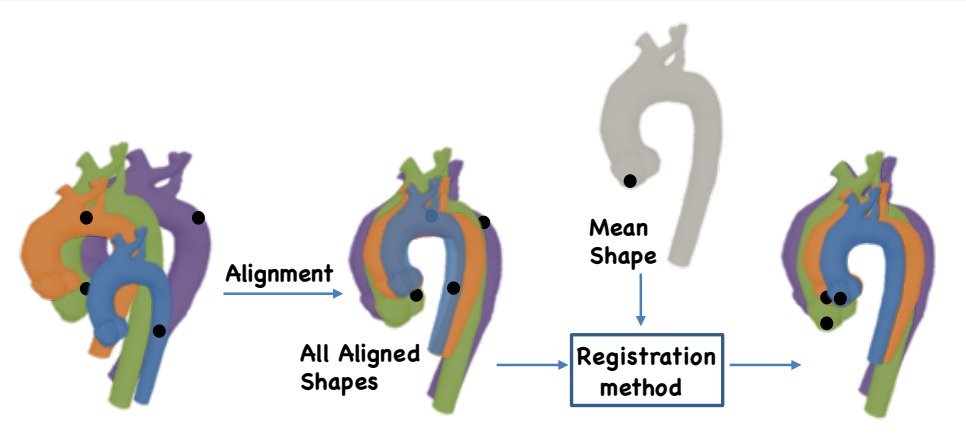
- **Rigid/Non-rigid Registration to establish correspondence**

  The final step before performing PCA was establishing correspondence across all point clouds, Sec 2.6. The primary goal is to align and deform the template (*source*) mesh to match each individual (*target*) mesh in the dataset with varying topologies and vertex counts, ensuring consistent topology and vertex correspondence post-deformation. A number of methods, some of which are rigid, such as ICP variants `Point-to-Point and Point-to-Plane` [30, 150] and `Fast Global Registration (FGR) with RANSAC` [246] and others non-rigid, such as CPD affine registration and CPD deformable registration [142], are used as baseline methods. Additionally, we used ICP `Point-to-Plane` to improve alignment precision by fine-tuning the transformation obtained through `RANSAC`. These methods were introduced in section 2.6. Implementations of CPD, `ICP` and `RANSAC` were used from `pycpd`[3] and `open3d`, respectively. Our developed deep learning

[3] https://github.com/siavashk/pycpd



(a) creating the template (mean shape)



(b) using the template for establishing correspondence

Figure 9.3: Template selection, alignment and registration. (a) template selection (b) how correspondence is established after registration. The same vertex (black) is at the same anatomical location as the template across all meshes

approach utilises `PyTorch3D` [163][4] and is compared to the baseline methods.

These registration methods aim to establish correspondence between the source and target points by aligning the source as closely as possible to the target. Rigid methods achieve this through a global transformation (only translation and rotation), while non-rigid methods like deformable CPD enable local adjustments to achieve point-to-point correspondence. Therefore, nearest neighbour matching is used to reorder the target points so that they align with the transformed source. This enables the target points to follow the source's vertex order, effectively transferring the source's topology onto the registered target geometry. Specifically:

1. For each point in the transformed source (output of `RANSAC`, `ICP`, or `CPD`), find the closest point in the target using a KD-tree for efficient nearest neighbour

[4] https://pytorch3d.org/

search.

2. Construct a reordered version of the target mesh by reindexing its vertices to match the source's vertex order, thereby ensuring consistent topology across meshes.

- **Proposed learning-based approach**
  The deformation process involves optimising vertex positions to minimise a composite loss function:

$$\mathcal{L}_{\text{total}} = w_{\text{chamfer}}\mathcal{L}_{\text{chamfer}} + w_{\text{edge}}\mathcal{L}_{\text{edge}} + w_{\text{normal}}\mathcal{L}_{\text{normal}} + w_{\text{Laplacian}}\mathcal{L}_{\text{Laplacian}} + w_{\text{curvature}}\mathcal{L}_{\text{curvature}} + w_{\text{IoU}}\mathcal{L}_{\text{IoU}} + w_{\text{dice}}\mathcal{L}_{\text{Dice}}$$

  where w are learnable weight coefficients for each loss component, optimised using log-space parameterisation to ensure numerical stability.

  – **Chamfer Distance** measures the average closest point distance between two point sets:

$$\mathcal{L}_{\text{chamfer}} = \sum_{\mathbf{v}\in\mathcal{V}_{\text{source}}} \min_{\mathbf{u}\in\mathcal{V}_{\text{target}}} \|\mathbf{v}-\mathbf{u}\|^2 + \sum_{\mathbf{u}\in\mathcal{V}_{\text{target}}} \min_{\mathbf{v}\in\mathcal{V}_{\text{source}}} \|\mathbf{u}-\mathbf{v}\|^2$$

  – **Edge Loss** enforces the preservation of mesh edges to maintain topology: $\mathcal{L}_{\text{edge}} = \sum_{(i,j)\in\mathcal{E}} \|\mathbf{v}_i - \mathbf{v}_j\|^2$, where $\mathcal{E}$ is the set of edges in the mesh.

  – **Normal Consistency** ensures that vertex normals remain consistent to preserve surface smoothness:
  $\mathcal{L}_{\text{normal}} = \sum_{i=1}^{N} \|\mathbf{n}_i^{\text{source}} - \mathbf{n}_i^{\text{deformed}}\|^2$, where $\mathbf{n}_i$ are the vertex normals.

  – **Laplacian Smoothing** minimises the Laplacian energy to smooth the mesh:

$$\mathcal{L}_{\text{Laplacian}} = \sum_{i=1}^{N} \|\mathbf{v}_i^{\text{deformed}} - \frac{1}{|\mathcal{N}(i)|}\sum_{j\in\mathcal{N}(i)} \mathbf{v}_j^{\text{deformed}}\|^2$$

  where $\mathcal{N}(i)$ denotes the neighbouring vertices of vertex i.

  – **Curvature Matching** computes the mean curvature at each vertex using the Laplacian operator and matches the curvature between source and target meshes:

$$\mathcal{L}_{\text{curvature}} = \text{Chamfer Distance}\left(\mathcal{H}^{\text{source}}, \mathcal{H}^{\text{target}}\right)$$

  where $\mathcal{H}$ represents the mean curvature values of vertices, which can be

approximated using the Laplacian operator as:

$$H_i = \frac{1}{d_i} \sum_{j \in N(i)} (\mathbf{v}_j - \mathbf{v}_i)$$

where: - H_i is the mean curvature at vertex i,
- d_i is the degree of vertex i (number of adjacent vertices),
- $N(i)$ is the set of neighbours of vertex i,
- $\mathbf{v}_j$ and $\mathbf{v}_i$ represent the positions of neighbouring vertex j and vertex i.

– **IoU and Dice Coefficient** are overlap metrics evaluating the similarity between the deformed source and target meshes:

$$\mathcal{L}_{\text{IoU}} = 1 - \frac{|\mathcal{V}_{\text{source}} \cap \mathcal{V}_{\text{target}}|}{|\mathcal{V}_{\text{source}} \cup \mathcal{V}_{\text{target}}|}, \quad \mathcal{L}_{\text{Dice}} = 1 - \frac{2|\mathcal{V}_{\text{source}} \cap \mathcal{V}_{\text{target}}|}{|\mathcal{V}_{\text{source}}| + |\mathcal{V}_{\text{target}}|}$$

The optimization aims to find vertex displacements $\Delta\mathcal{V}$ that minimize the total loss:

$$\Delta\mathcal{V}^* = \arg\min_{\Delta\mathcal{V}} \mathcal{L}_{\text{total}}(\mathcal{V} + \Delta\mathcal{V}, \mathcal{F}, \mathcal{V}_{\text{target}}, \mathcal{F}_{\text{target}})$$

Using Adam Optimiser with log-space weight initialisation and a 0.01 learning rate for weights and 1.0 for vertex displacement:

$$\Delta\mathcal{V}_{t+1} = \Delta\mathcal{V}_t - \eta \nabla_{\Delta\mathcal{V}} \mathcal{L}_{\text{total}}$$

The optimisation runs for a predefined number of epochs or until early stopping criteria are met. For each epoch:

1. Compute the total loss $\mathcal{L}_{\text{total}}$ with learnable weights.
2. Perform backpropagation to compute gradients: $\nabla_{\Delta\mathcal{V}} \mathcal{L}_{\text{total}}$.
3. Update the vertex displacements and weights using Adam optimiser.
4. Early stopping if there is no improvement after 200 consecutive iterations.

By deforming the template to match each target mesh, we ensure that all meshes share the same vertex structure, which is essential for subsequent shape analysis.

- **Evaluation** All registration methods were evaluated by calculating RMSE, Hausdorff, IoU, Dice, and the average distance between the target and deformed source mesh (template mesh). Details of evaluation methods can be found in section 2.7.

9.4 Identifying aortic shape variations

PCA was mentioned in section 2.2 as a method for linear dimensionality reduction. Here, PCA is used to identify key modes of aortic shape variation and visualise them in 3D. The input data consists of 599 registered meshes, each represented by 1000 vertices with x, y, and z coordinates, resulting in a total of 3000 features per subject. Singular Value Decomposition (SVD) is used to project the input matrix of size 599x3000 to a lower-dimensional space for efficient computation.

The components are sorted in descending order of explained variance, with the first component capturing the largest proportion of variance in the dataset. We analyse principal components that collectively explain over 95% of the total variance. The mean shape of the cohort is visualised as a 3D mesh, along with the shape variations corresponding to positive and negative displacements of two standard deviations along the first few principal components. Since the point clouds were registered before applying PCA, we maintain the connectivity (faces) of the original template mesh to visualise the shape variations. These varying meshes are saved as `.vtk` files, enabling interactive 3D visualisation in Paraview[5].

Additionally, we compare the use of Umeyama rigid and similarity transformations [206] to the registered meshes before applying PCA to better understand sources of variation beyond overall size. The similarity transformation includes a global isotropic scaling factor, allowing us to factor out the overall size effects and focus on capturing localised shape variations. We analyse the PCA results from both transformations, examining the variance explained by each principal component and the nature of the shape deformations. Visualising and quantifying these differences allows for a more precise understanding of aortic shape variation in the cohort.

[5] https://www.paraview.org/

9.5 Results

The first set of results, presented in Section 9.5.1, evaluates the performance of the various registration methods. Subsequently, the meshes registered using the best-performing method are analysed using PCA, with the results detailed in Section 9.5.2.

9.5.1 Mesh registration

Rigid registration methods, such as `ICP` variants, `RANSAC` and `ICP+RANSAC`, showed limitations in handling non-rigid deformations, often resulting in incomplete template coverage and biased alignment. These methods are constrained by their reliance on rigid transformations, which fail to accommodate complex anatomical variations. Their outlier rejection mechanisms, while intended to improve robustness, often lead to incomplete template coverage and biased alignment. This issue is particularly pronounced in cases of partial overlap or poor initial alignment, where regions with better correspondence dominate the transformation, resulting in artificial distortions or "holes" in the final output.

When handling partial overlaps, these rigid methods prioritise locally optimal alignments at the expense of global geometric consistency. This is particularly evident in cases of poor initial alignment, where `ICP+RANSAC` produces distorted transformations due to its inability to accommodate local deformations. The observed differences post-transformation primarily stem from incomplete or partial alignment rather than true non-rigid transformation, resulting in regions that either closely match the target or appear entirely misaligned due to a lack of correspondence.

In contrast, non-rigid approaches, including CPD (affine and deformable) and our proposed deep learning-based method, demonstrated superior performance, as shown in Figure 9.4. CPD, in both its affine and deformable variants, demonstrated better management of global and local deformations, with the deformable variant showing particular improvement in preserving geometric features while allowing non-rigid transformation. Our deep learning approach, leveraging a composite loss function, outperformed all classical methods across all metrics, preserving geometric integrity and achieving accurate registration. This highlights the potential of learning-based techniques to address the limitations of traditional methods. The visual comparison in Fig. 9.5 reinforces these findings.

9.5.2 Shape analysis using PCA

Each principal component explains a percentage of the total variance present in the cohort regarding the 3D shape of the aorta. These percentages are sorted in descend-

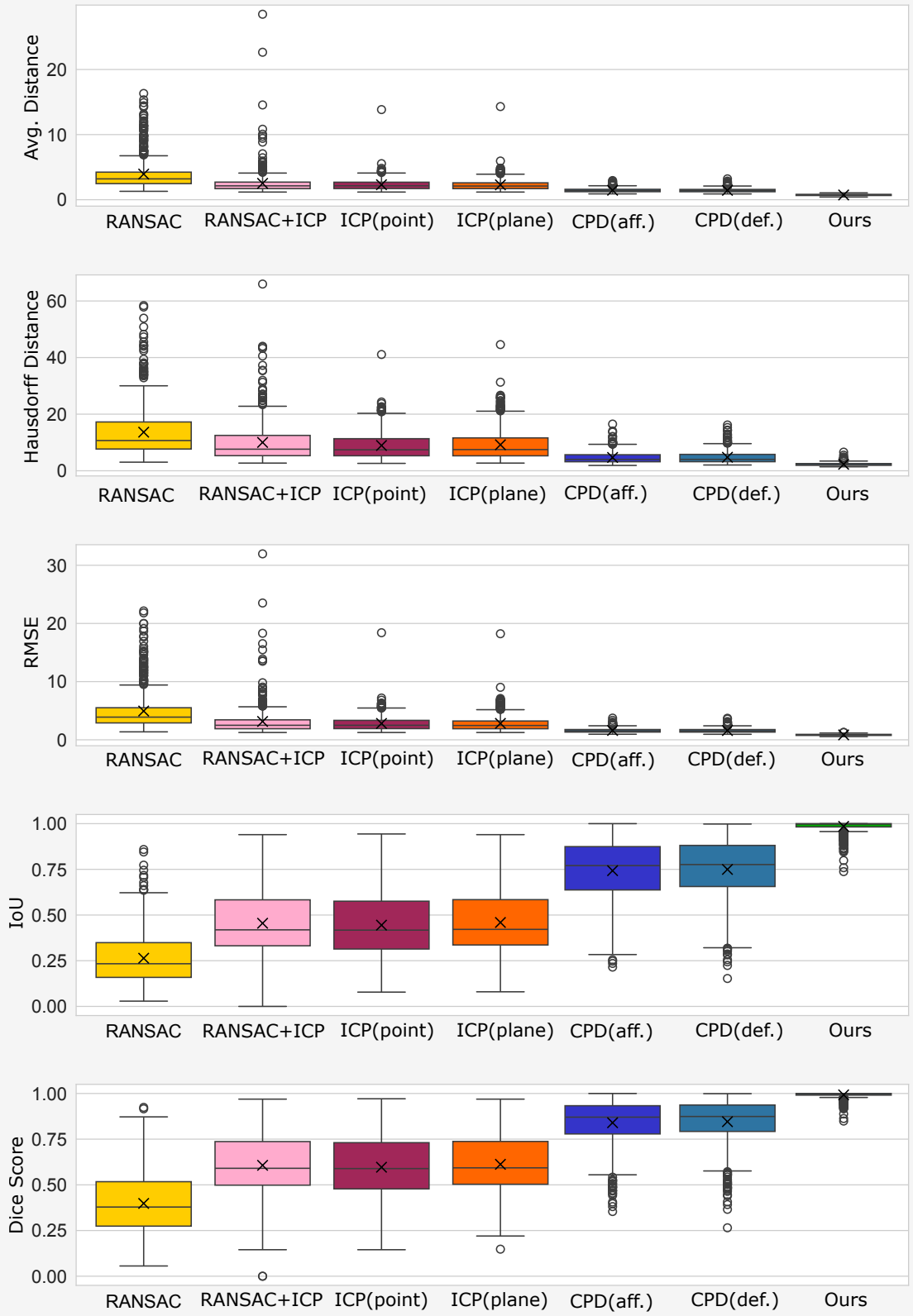
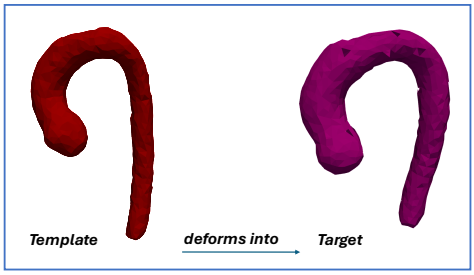


Figure 9.4: Comparison of rigid and non-rigid registration methods using different evaluation metrics



Method	Avg dist ↓	Haussdorf dist ↓	RMSE ↓	IoU ↑	Dice ↑
RANSAC	4.29	18.82	5.48	0.17	0.29
RANSAC+ICP	2.42	7.44	2.81	0.32	0.48
ICP (point)	2.30	6.82	2.63	0.34	0.51
ICP (plane)	2.31	7.63	2.69	0.35	0.52
CPD (affine)	1.74	9.30	2.07	0.60	0.75
CPD (def.)	1.73	8.61	2.06	0.60	0.75
Ours	**0.78**	**2.59**	**0.92**	**0.99**	**0.99**

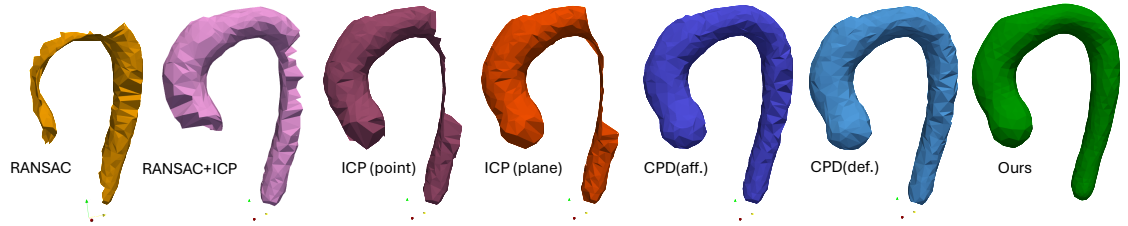
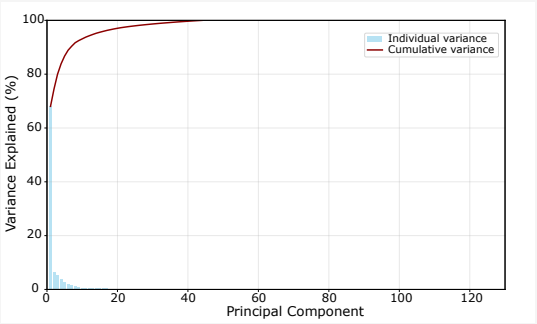
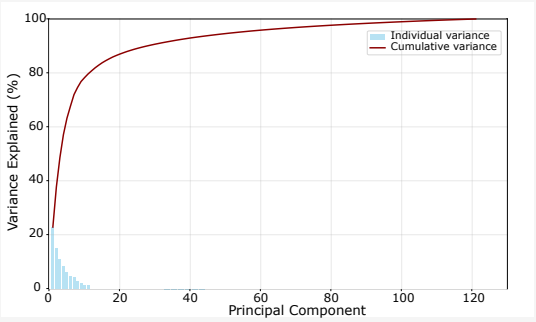


Figure 9.5: A random target mesh was selected to visualise how the template deformed under each registration method

ing order to identify which mode of variation contributes the most. The percentages are shown as vertical bars, and their accumulative sum is shown by a curve in Fig. 9.6 and listed in Table 9.1.



(a) Umeyama rigid



(b) Umeyama similarity

Figure 9.6: Percentage of variance explained by each principle component and their accumulated curve

The first six modes are visualised as 3D meshes in Fig. 9.7, demonstrating the primary modes of variation. The first PCA mode captures overall size changes under both the Umeyama rigid and similarity transformations. However, it explains a significantly larger proportion of variance—68.0% with the rigid transformation compared to 22.9% with the similarity transformation, a threefold difference, Fig. 9.6 and Table 9.1. This redistribution results in subsequent modes explaining relatively more variance under the similarity transformation, with differences around twofold for modes two to six. Despite these shifts, the interpretations of the PCA modes remain consistent across both transformations for all visualised modes. These findings suggest

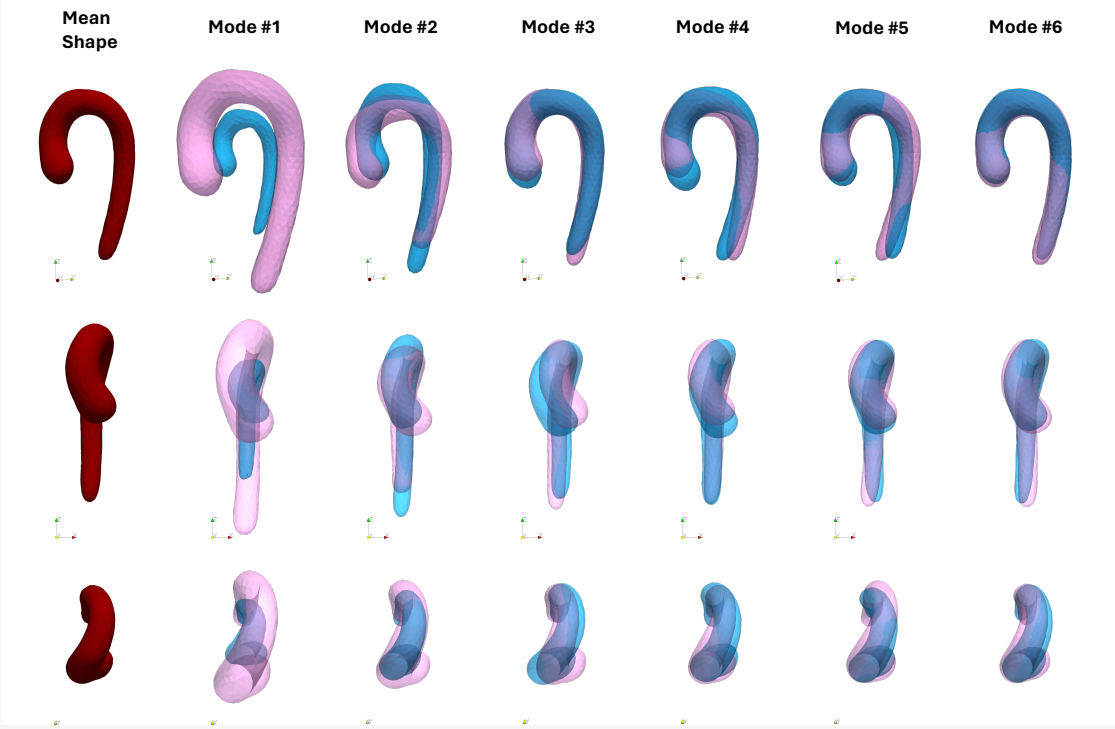
PCA Mode	variance explained (rigid transformation)	variance explained (similarity transformation)
Mode 1	68.0% (cumulative: 68.0%)	22.9% (cumulative: 22.9%)
Mode 2	6.5% (cumulative: 74.5%)	14.9% (cumulative: 37.7%)
Mode 3	5.5% (cumulative: 80.0%)	10.9% (cumulative: 48.6%)
Mode 4	3.8% (cumulative: 83.8%)	8.5% (cumulative: 57.1%)
Mode 5	2.9% (cumulative: 86.7%)	6.1% (cumulative: 63.2%)
Mode 6	2.2% (cumulative: 88.9%)	4.5% (cumulative: 67.7%)
Mode 7	1.5% (cumulative: 90.4%)	4.3% (cumulative: 72.0%)
Mode 8	1.3% (cumulative: 91.7%)	2.6% (cumulative: 74.6%)
Mode 9	0.8% (cumulative: 92.5%)	2.2% (cumulative: 76.8%)
Mode 10	0.6% (cumulative: 93.1%)	1.5% (cumulative: 78.3%)

Table 9.1: Variance explained by the top 10 principal components for aortic shape analysis after applying Umeyama rigid and similarity transformations separately.

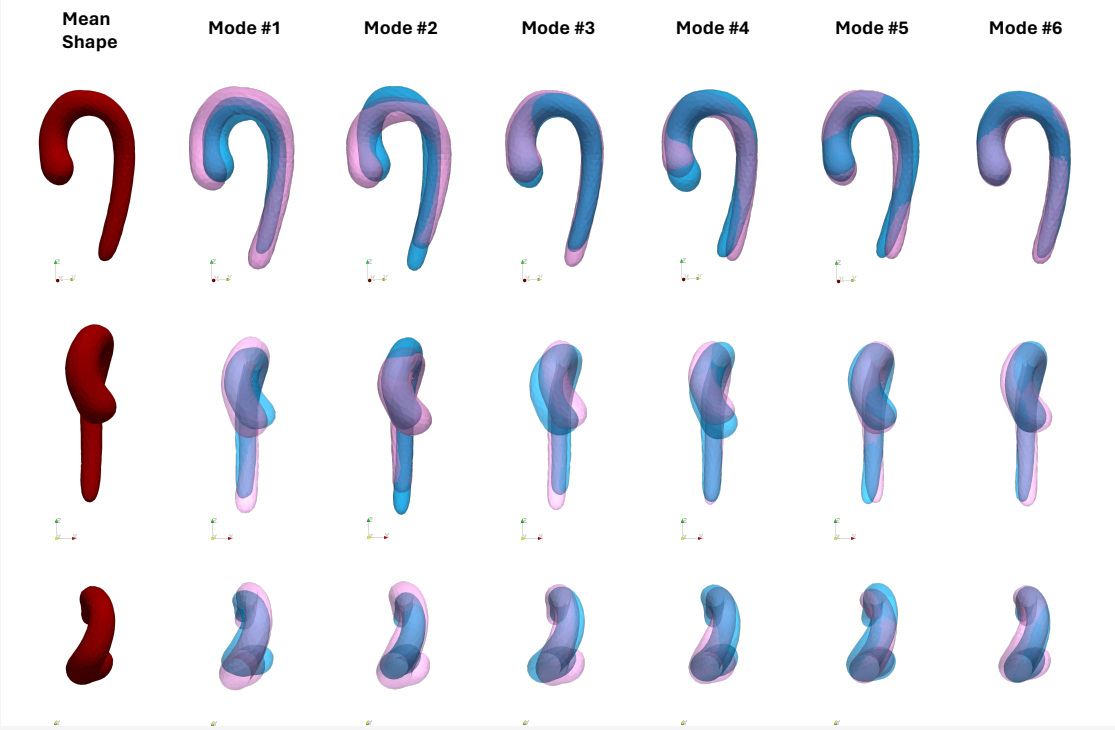
that both transformations highlight comparable aortic shape characteristics, with the similarity transformation emphasising finer variations across multiple modes.

Below is a list of the shape or size features described by each of the first six modes, respectively, from Fig. 9.7:

1. **overall aortic size (scaling)**
2. **change in aortic arch height and width**, bringing the ascending and descending aorta closer together as the aortic arch height increases, decreasing the aortic arch width.
3. the **orientation of the ascending aorta**, tilting sideways, as can be seen across all three views.
4. the relationship between aortic arch curvature and the lengthening or shortening of the ascending aorta, which is more visible in the first two views.
5. related to the aortic arch shape but without substantial changes to the aortic length.
6. related to minor changes in the in-plane orientation of the aortic arch in relation to the ascending and descending aorta.



(a) Umeyama rigid before applying PCA



(b) Umeyama similarity before applying PCA

Figure 9.7: The panels from left to right show the average shape in bright red, followed by the top 6 modes of variation. The shapes after subtracting and adding two standard deviations of each mode to the average shape (in pink and blue, respectively). The three rows show the same shapes, just from different viewing angles

9.6 Conclusion

While computationally efficient and suitable for simple alignments, rigid methods show fundamental limitations in handling non-rigid deformations. This limitation becomes particularly apparent in partial overlap cases, where rigid methods often create incomplete or distorted transformations. The superior performance of non-rigid methods, especially the deep learning approach, highlights the importance of local deformation capability in achieving accurate registration.

For scenarios requiring rigid alignment, point-to-point ICP remains a reliable choice over other rigid registration methods and can serve as a suitable method for coarse alignment. However, it should be followed by more sophisticated, non-rigid methods for final refinement. When dealing with non-rigid cases, the deep learning approach or deformable CPD should be favoured.

We also demonstrate the effectiveness of PCA in characterising aortic shape variations across a large population. The analysis revealed distinct patterns in how variance is distributed across principal components under different transformation approaches. With rigid transformation, the first mode dominated, explaining 68.0% of total variance, primarily capturing global size differences. In contrast, the similarity transformation produced a more distributed variance pattern, with the first mode accounting for only 22.9%, suggesting a size-independent representation of shape variations.

The cumulative variance explained by the first ten modes reached 93.1% under rigid transformation and 78.3% under similarity transformation, indicating that a relatively small number of principal components can effectively characterise the majority of aortic shape variations in the population. This finding has important implications for clinical applications and future research, potentially simplifying the quantitative assessment of aortic morphology and enabling more targeted analysis of shape-based risk factors in aortic diseases. Future work could focus on correlating these identified shape modes with clinical outcomes and investigating their potential as predictive markers for aortic pathologies.

Chapter 10

Characterising Aortic Shape Variability

Structural abnormalities of the aorta can impair its functionality, while functional impairments may, in turn, induce structural changes. Moreover, variations in aortic morphology and function may be influenced by demographic factors, underscoring the need for a comprehensive analysis of these interrelations. To address this gap, it is essential to characterise the aorta's phenotype, its determinants, and its implications for function, both in health and disease. Establishing a benchmark for aortic shape and function is crucial for identifying pathological deviations.

This study presents a comprehensive analysis of aortic shapes reconstructed from Cardiac Magnetic Resonance (CMR) images of a cohort of 599 healthy subjects. PCA is used to quantify healthy shape variability, enabling an investigation of dominant modes of variation. We constructed a correlation matrix to identify the relationship between these dominant PCA modes and target variables related to the cohort's demographics, aortic size, left ventricular function, aortic flow, and blood pressure. These identified strong correlations are used to define regression tasks, where PCA modes serve as predictors for the associated target variables. We also compare the predictive performance of these linear regression models to our bespoke GNN model.

This analysis establishes a benchmark for the shape of the healthy aorta and provides insight into the factors influencing its morphological variability. This work contributes to a more comprehensive understanding of aortic shape, size, and function by leveraging dimensionality reduction techniques and deep learning approaches.

10.1 Introduction

Medical images encode a massive amount of information of high clinical value encoded in them. Hence, image analysis techniques, cardiac atlases, and computational models are currently under active development to capture the complexity of the cardiovascular system. Advanced shape analysis techniques are promising for enhancing patient-specific clinical applications, including surgical pre-planning, improved intra-operative navigation, optimised medical device design, and more accurate outcome prediction.

Traditional methods of analysing shapes involve computing hand-crafted feature descriptors and extracting a limited number of predefined manual measurements. 2D measurements, such as diameters from oblique images, usually recorded by experts, can be time-consuming, irreproducible, and often not as accurate as 3D measurements [221]. The literature lacks a comprehensive study analysing the 3D thoracic aorta shape, relating its structure to function using feature learning. Instead, some research relied on statistical shape modelling (SSM) for holistically analysing the aortic shape [21, 74, 200]. Rodero et al. extended the application of SSM to link structure to function [169].

Others investigated statistical approaches to predict the size of the aorta at different locations using factors such as gender, age, and body surface area (BSA). Many regression models have been studied, including linear, logarithmic, quadratic, polynomial, and cubic functions. Linear regression models consistently emerged as the most effective statistical approach for representing aortic dimensions [23, 154, 170]. Among predictors, BSA demonstrated the strongest correlation with aortic dimensions in most studies [26, 154, 170, 210, 220], though Roman et al. found age to have a higher correlation [170]. Additionally, gender differences add complexity, with aortic root dimensions generally being smaller in women compared to men of similar age and body size [129, 211]. Age-related changes show aortic diameters increase over time in both genders, though rates may differ [100, 220].

The complex interplay of factors affecting aortic dimensions and shape, including BSA, gender, age, and segment-specific variations, underscores the need for advanced analytical approaches. Therefore, we investigate using statistical shape models, linear regression models and our multi-graph architecture for aortic shape analysis. In addition to commonly studied factors such as age, sex, and BSA, we also include factors related to LV function, aortic flow, and blood pressure to better understand the relationship between aortic structure and function.

10.2 Methods

10.2.1 Imaging dataset and pre-processing

The aorta was segmented from the E-HVols imaging dataset (see Section 3.1.5) using nn-UNet. Then we extracted the aortic shapes using Marching Cubes before applying our 3D surface reconstruction pipeline including (1) downsampling meshes to a fixed number of vertices, (2) recreate the surface using the new number of vertices and (3) mesh repair including filling holes, removing non-manifold edges, smoothing and other, please see Section 9.2.2 for details. Finally, to establish correspondence across the dataset, we use our developed deep-learning approach presented in Chapter 9, which allowed us to use PCA for identifying the anatomical modes of variation of the aorta.

10.2.2 Target variables

Left ventricular function related parameters:

First **LV mass, end-systolic volume (ESV) and end-diastolic volume (EDV)** are obtained by tracing the endocardium and epicardium from short-axis images of the LV, slice by slice using `Segment software (Medviso AB, Lund, Sweden)`, see Fig. 10.1. Subsequent parameters were derived from these measurements:

- **Stroke Volume (SV)**: The blood volume ejected from the ventricle during systole, calculated as the difference between EDV and ESV.
- **Ejection Fraction (EF)**: The percentage of blood leaving the heart with each contraction, computed as EF = SV/EDV.
- **Ventriculo-Arterial Coupling (VAC)**: Represents how efficiently the heart's pumping action matches arterial resistance. It is approximated using the volume method: VAC = (EDV-ESV)/ESV [227].

Pulse Wave Velocity (PWV):

PWV is calculated using `Segment software (Medviso AB, Lund, Sweden)`, based on the formula: PWV $= \Delta D/\Delta T$, (m/s) where ΔD represents the aortic path length between the mid-aortic valve and mid-descending aorta, and ΔT is the time delay between the pulse wave's arrival at these levels [90].

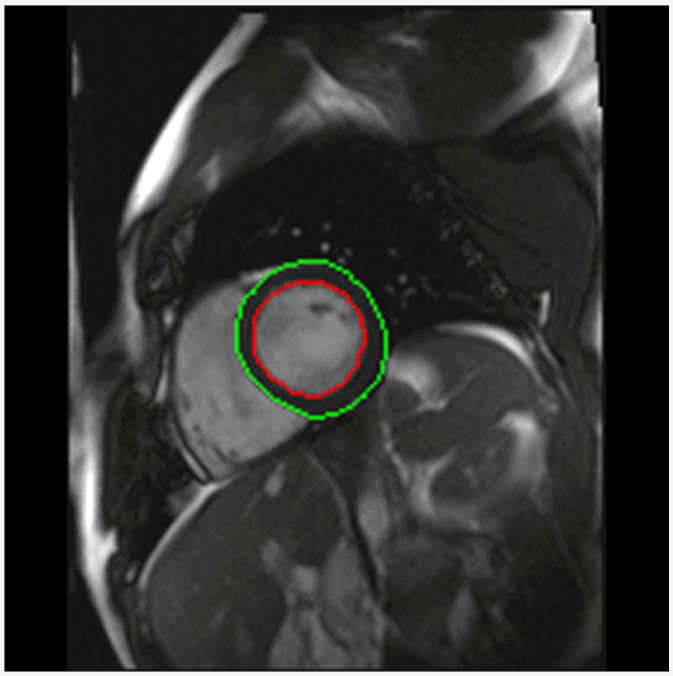


Figure 10.1: Slice from short-axis stack of the LV. Traced endocardium and epicardium for LV function calculations

Velocity Mapping Phase contrast MRI produces two types of images: magnitude and phase images, as shown in Fig. 10.2. In magnitude images, brighter areas indicate stronger signals from the spinning magnetisation of tissue inside the MRI scanner. This magnetisation is a vector, meaning it has both size and direction.

The MRI scanner detects whether the signal is ahead or behind a reference oscillator in the machine. The phase of the signal represents its direction, which helps determine the direction of blood flow. This is useful because magnetic field gradients can cause phase shifts, allowing MRI to encode position information [63]. The measured velocity estimates the true average velocity within a voxel, given in cm/s, Fig. 10.2.

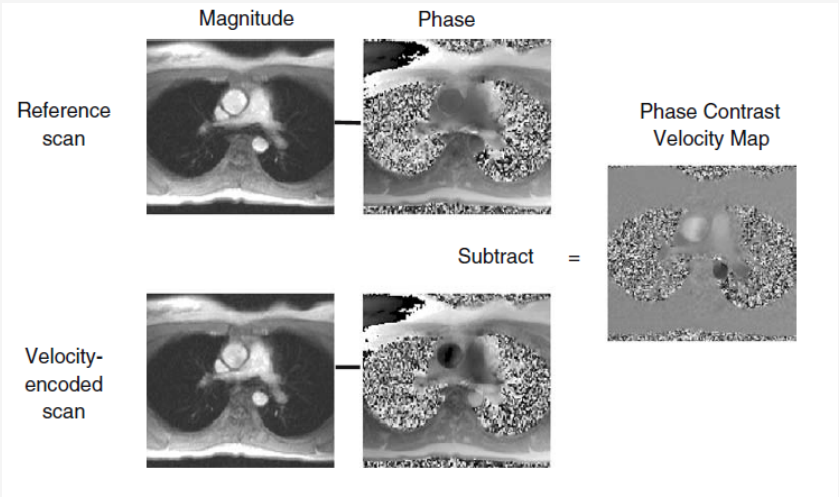


Figure 10.2: Magnitude and phase images from the reference scan (top row) and velocity-sensitive scan (bottom row). Subtracting the phase images gives the velocity map [63].

Mean flow:

A MATLAB code developed by Torii et al. [203] was used for velocity mapping. The region of interest (ROI) was defined by manually placing points along the contour of the aortic root cross-section for each slice. The velocities within the ROI were averaged to obtain mean flow values.

Aortic size-related parameters:

A tool developed in `MATLAB (MathWorks, Natick, Massachusetts)` extracted aortic sizing parameters automatically by locating nine anatomical landmarks at the annulus, STJ and extreme bulging points per aortic sinus [184], Fig. 10.3. The centreline was generated using Mimics Research version 19.0 (Materialise, Leuven, Belgium). A

plane was fitted to each set of three landmarks or oriented orthogonally to the centre-line. Then, the mesh-plane intersection contours are extracted, and a circle of best-fit is used to estimate the diameter of the cross-section [114].

The sinuses of the aortic root are separated using cutting planes that rotate interac-
2870 tively about the centreline of the aortic root. Sinus volumes were then calculated using the α-shapes algorithm [51].

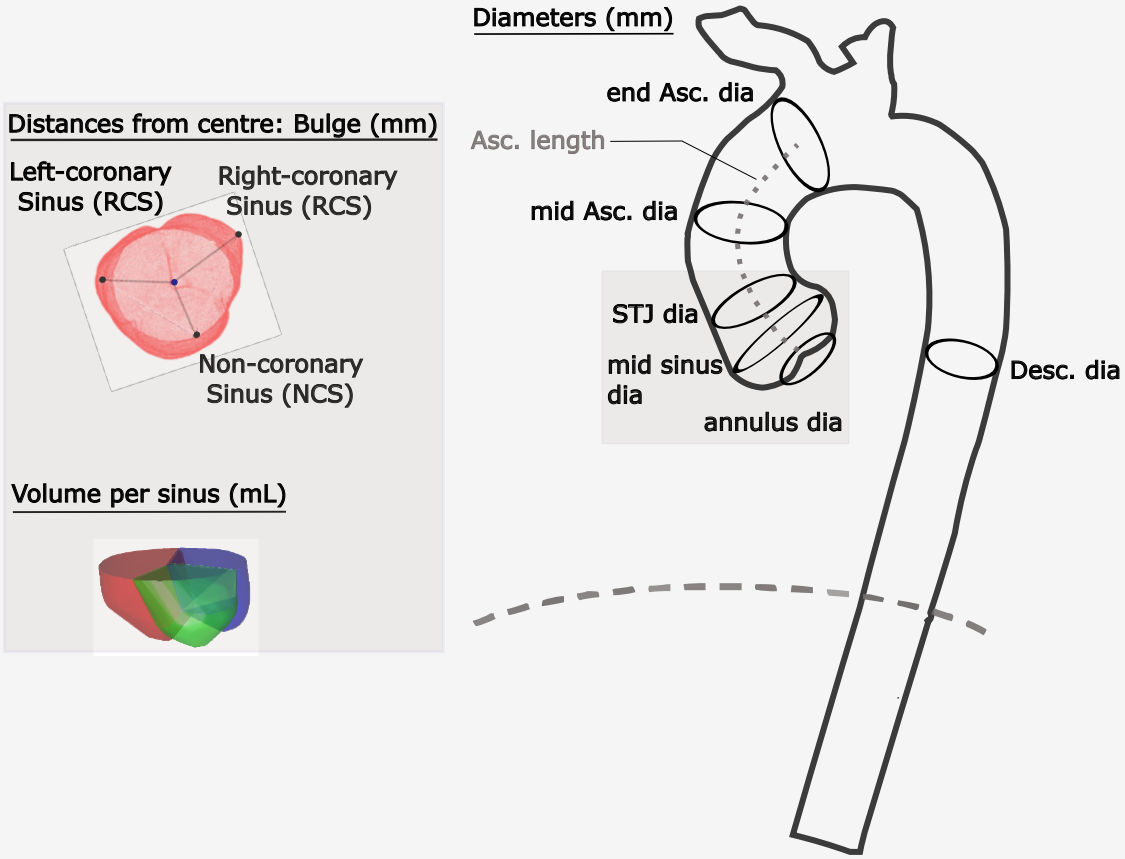


Figure 10.3: Size-related parameters. Left: size of the aortic root sinuses, including volume per sinus and the distance from the centre to each bulging landmark. Right: Diameters recorded along the centreline using circle of best-fit [a] and length of the centreline along the ascending aorta, denoted as Asc. length.

[a] annulus diameter, diameter at maximum bulges level (mid sinus dia), sino-tubular junction diameter (STJ dia), diameter at mid ascending aorta (mid Asc. dia), diameter at end ascending aorta before the first bifurcation (end Asc. dia) and diameter at mid descending aorta (Desc. dia)

10.2.3 Data exploration

To find correlations between aortic shape and available target data, we use the projections of shape data onto the top six principal components obtained through PCA.
2875 Further details are provided in Section 9.4. We calculated a correlation matrix to explore the relationships between target variables and PCA modes. This matrix provides

a visual overview of how strongly each target variable correlates with the six PCA modes. This visualisation helps identify which PCA modes might be most relevant for predicting specific targets.

10.2.4 Power analysis

We conducted a power analysis to determine the minimum sample size required for our multiple linear regression analysis. This analysis was crucial to ensure that our study had sufficient statistical power to detect meaningful relationships between the predictor variables (six PCA modes describing aortic shape) and our target variables. We based our power analysis on established guidelines by Cohen et al. [35].

The significance level alpha was set at 0.05 and we aimed for a power of 80%, which are commonly accepted standards [55]. We anticipated a medium effect size, corresponding to an R^2 value of 0.13, which translates to a Cohen's f^2 of 0.1494 [35].

10.2.5 Linear regression & baselines

To establish a baseline, we use linear regression on projections of shape data onto the top six principal components. This approach allowed us to assess the predictive potential of each target label in a lower-dimensional space and set a benchmark for more advanced non-linear models, e.g. our multi-graph architecture.

10.2.6 Single and multi-target prediction using our GNN

Our multi-graph architecture processes the aortic meshes to predict target variables. The model uses a shared backbone for feature extraction and separate output layers for each target variable. This design allows the model to learn shared representations of the aortic shape while tailoring predictions to individual targets. It can handle both single-target prediction, where one target variable is predicted per forward pass, and multi-target prediction, where multiple target variables are predicted simultaneously in a single pass.

10.3 Experiments

We used only subjects with available target data, leading to a different number of samples per target label. Our multi-graph architecture uses the same data splits to predict selected target variables as the linear regression models.

1. **Cross-Validation with 5 Folds:**
   To ensure robust evaluation, we performed 5-fold cross-validation, splitting the data into five equal parts. Four parts were used for training, and one was held out for testing in each fold, rotating assignments to provide averaged performance metrics and mitigate overfitting.

2. **Mode combinations as features:**
   Each target was predicted using linear regression based on individual and combined principal component projections. We tested combinations ranging from single modes to all six principal components.

3. **Median predictor baseline:**
   As a reference point, we calculated the Mean Absolute Error (MAE) for a baseline model. This model used the median of the target values in the training set as a constant prediction for the test set in each fold. Comparing this baseline MAE to the linear regression MAE allowed us to assess whether our model provided meaningful predictive value.

4. **Comparison of Mesh Processing and Model Architectures for our GNN:**
   We first compare the use of the unprocessed meshes produced by Marching Cubes versus the pre-processed meshes described in Section 10.2.1. We also compare the use of GCNConv and SplineConv, as these have yielded better performance in our previous comparative study compared to GraphConv from Chapter 4. Due to the small dataset size, an augmentation strength of 0.5 was used.

5. **Normalization for the multi-target prediction Model:**
   To account for differences in scale among target variables, we normalise the outputs and ground truth values using the per-batch standard deviation before computing the Mean Squared Error (MSE) loss. This ensures that targets with larger numerical ranges do not disproportionately influence training. These experiments are repeated without normalisation to study the effect of normalisation on model performance.

Model performance was evaluated using Mean Absolute Error (MAE) and R^2 metrics for each target variable.

10.4 Results

10.4.1 Power analysis

Fig. 10.4 demonstrates how statistical power increases with sample size for different effect sizes, providing a comprehensive understanding of the power dynamics in our study design. We determined that a sample size of 98 or more is required for a medium effect size to achieve sufficient statistical power of 0.8 at a significance level of 0.05. This finding underscores the importance of ensuring an adequate sample size to detect meaningful relationships between aortic shapes and target variables.

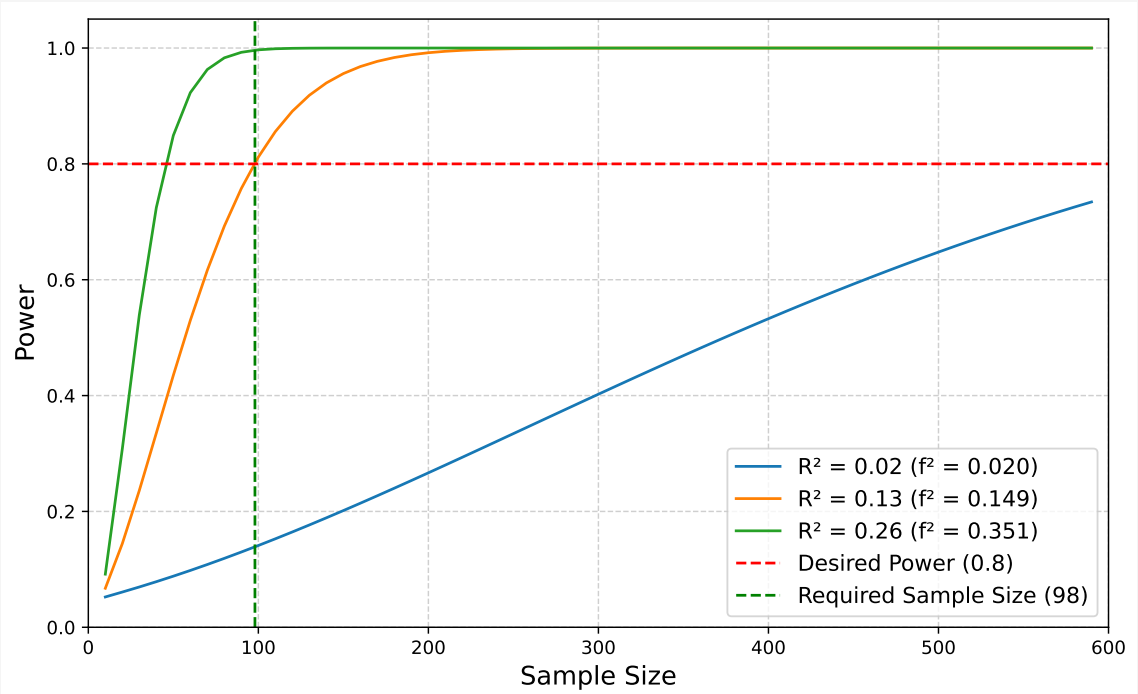


Figure 10.4: Power Curves for Multiple Linear Regression. The curves show the relationship between sample size and statistical power for small (R^2 = 0.02), medium (R^2 = 0.13), and large (R^2 = 0.26) effect sizes. The red dashed line marks the desired power of 0.80, and the green dashed line indicates the minimum required sample size of 98.

10.4.2 Target variables and PCA projections relationships

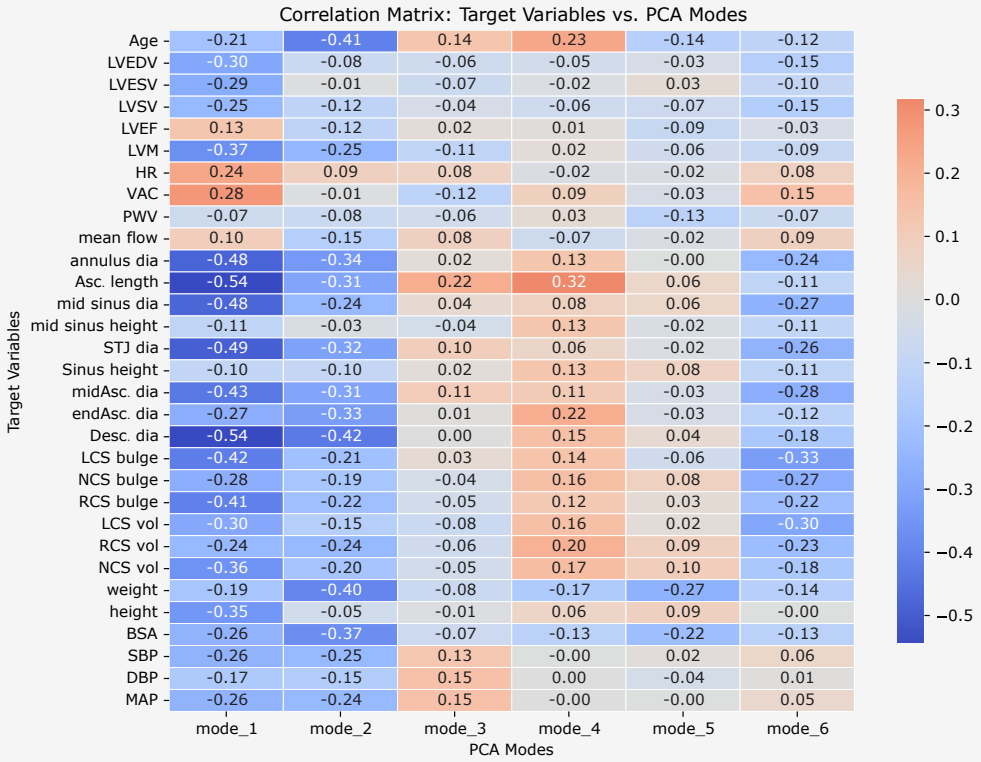
Similar patterns of correlation were observed in Fig. 10.5a and Fig. 10.5b, although with slight differences in magnitude.

Fig. 10.5 shows that the strongest correlations ($|r|$>0.2) were found between two or more PCA modes and age, LVM, BSA and weight, systolic blood pressure (SBP), and mean arterial pressure (MAP) and all aortic size-related variables except for sinus

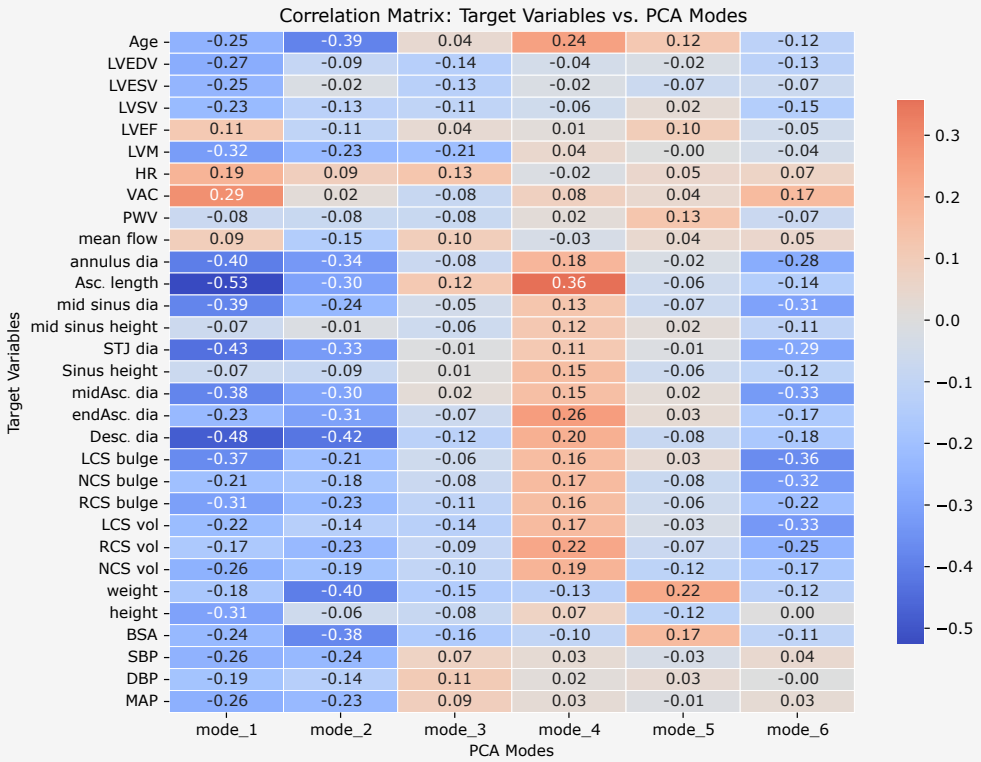
height and mid-sinus height. These findings indicate that the aortic shape potentially influences several target variables. This is summarised in Table 10.1.

Table 10.1: Target variables showing strong correlation ($|r| > 0.2$ in ≥2 PCA modes) with shape descriptors derived from rigid and similarity registrations.

Category	Target Variable	Rigid	Similarity
Demographics	Age	✓	✓
	Weight	✓	✓
	Height	✗	✗
	BSA	✓	✓
LV Function	LVEDV	✗	✗
	LVESV	✗	✗
	LVSV	✗	✗
	LVEF	✗	✗
	LVM	✓	✓
	HR	✗	✗
Hemodynamics	VAC	✓	✓
	PWV	✗	✗
	Mean flow	✗	✗
Aortic Size (Root & Ascending)	Annulus Diameter	✓	✓
	Asc. length	✓	✓
	Mid sinus diameter	✓	✓
	Mid sinus height	✗	✗
	STJ diameter	✓	✓
	Sinus height	✗	✗
	Mid Asc dia	✓	✓
	End Asc dia	✓	✓
	Desc dia	✓	✓
Aortic root bulge & Volume	LCS bulge	✓	✓
	NCS bulge	✓	✓
	RCS bulge	✓	✓
	LCS vol	✓	✓
	RCS vol	✓	✓
	NCS vol	✓	✓
Blood Pressure	SBP	✓	✓
	DBP	✗	✗
	MAP	✓	✓



(a) rigid Umeyama.



(b) similarity Umeyama.

Figure 10.5: Comparison of correlations between PCA modes and target variables after applying different Umeyama methods.

Correlation coefficient r measures the strength and direction of a linear relationship between two variables, ranges from -1 to 1.
Coefficient of determination R^2 measures how well a regression model (which can include multiple variables, e.g. PCA modes) explains the variation in the dependent variable (target variables we are trying to predict). It tells you the proportion of the variance in the target variable that is predictable from the PCA modes.

10.4.3 Linear regression models

The results of the linear regression models in Tables 10.2 and 10.3 show that the same set of target variables with strong correlations as in Fig. 10.5, where correlation coefficients $|r|>0.2$ between two or more PCA modes and targets also yielded high R^2 values >0.2 in the regression analysis. However, notable exceptions include LVM, SBP, VAC and specific size-related parameters, particularly aortic root sinus dimensions and end of ascending aorta diameter yielded lower R^2 values, ranging between 0.1 and 0.2.

This discrepancy suggests that while linear regression can capture some relationships between PCA modes and target variables, it may fail to account for more complex interactions or non-linear dependencies in the data. Therefore, these results underscore the need for advanced modelling techniques that capture non-linear relationships and interactions between representations of aortic shapes and target variables, such as GNNs.

10.4.4 Effect of mesh processing on GNN performance

Fig. 10.6a shows that the mesh registration and post-processing steps from Chapter 9 lowered the AUC for the sex classification task from 0.91 to 0.88. The superior performance of raw meshes suggests that certain geometric features critical for prediction may be smoothed out or lost during registration and post-processing steps. Therefore, the meshes produced from marching cubes were used to train and test our GNN models. Additionally, SplineConv was selected for the convolutional layers because it outperformed the model using GCNConv as shown in Fig. 10.6b

10.4.5 GNN performance predicting selected target variables

The GNN model's performance for predicting selected target variables is summarised in Table 10.4. GNNs demonstrated mixed results depending on the target variable compared to linear regression performance. For instance, GNNs underperformed significantly when predicting age ($R^2 = -0.276, MAE = 6.974$), BSA ($R^2 = 0.195, MAE = 0.163$) and ascending aorta length

Table 10.2: MAE and R^2 for regression models using different combinations of PCA modes after rigid Umeyama to predict available target variables

Target	N	Mod Comb (Lowest MAE)	R^2	MAE	Mod Comb (Highest R^2)	R^2	MAE	Baseline MAE
Age(years)	599	1,2,3,4,5,6	0.294	5.845	1,2,3,4,5,6	0.294	5.845	6.534
LVEDV(mL)	583	1,2,3,4,5,6	0.100	22.705	1,3,4,5,6	0.100	22.734	24.377
LVESV(mL)	583	1,3	0.081	12.395	1,3,6	0.086	12.406	13.032
LVSV(mL)	583	1,2,3,5,6	0.084	13.955	1,2,4,5,6	0.094	13.983	14.759
LVEF(%)	582	1,2,5	0.021	4.827	1,2,5	0.021	4.827	4.932
LVM(g)	390	1,2,3	0.160	22.165	1,2,3,6	0.164	22.177	25.088
HR(bpm)	372	1	0.018	9.057	1,3	0.019	9.065	9.353
VAC	162	1,4	0.057	0.551	1,4	0.057	0.551	0.581
PWV(m/s)	163	2,3,4,5	-0.043	0.842	2,6	-0.036	0.850	0.789
Mean flow(mL/s)	318	1,2	-0.032	17.212	1	-0.031	17.323	17.434
Annulus dia.(mm)	163	1,2,6	0.264	1.861	1,2,6	0.264	1.861	2.362
Asc length(mm)	186	1,2,3,4	0.468	4.886	1,2,3,4	0.468	4.886	6.986
Mid sinus dia(mm)	186	1,2,6	0.277	2.305	1,2,6	0.277	2.305	2.773
Mid sinus height(mm)	186	1,4	-0.070	2.554	4	-0.065	2.581	2.539
STJ diameter(mm)	186	1,2,6	0.347	1.900	1,2,6	0.347	1.900	2.451
Sinus height(mm)	186	2,4,5	-0.011	3.268	1,2,4	0.006	3.289	3.354
mid Asc dia(mm)	186	1,2,3,4,6	0.247	1.885	1,2,3,6	0.251	1.899	2.210
end Asc dia(mm)	186	1,2,4	0.142	2.525	1,2,4	0.142	2.525	2.690
Desc dia(mm)	186	1,2,3,4,6	0.446	1.500	1,2,4,6	0.450	1.515	2.168
LCS bulge (mm)	186	1,2,4,6	0.256	1.592	1,2,4,6	0.256	1.592	1.925
NCS bulge (mm)	186	1,2,5,6	0.129	1.826	1,2,4,6	0.136	1.844	1.968
RCS bulge (mm)	186	1,2,3,4,6	0.168	1.428	1,2,4,6	0.176	1.430	1.645
LCS vol (mL)	186	1,2,3,4,6	0.142	1.188	1,2,3,4,6	0.142	1.188	1.323
RCS vol (mL)	186	1,2,3,4,6	0.139	1.358	1,2,3,4,6	0.139	1.358	1.465
NCC vol (mL)	186	1,2,3,4,5,6	0.159	1.254	1,2,3,4,6	0.170	1.254	1.382
Weight(Kg)	599	1,2,3,4,5,6	0.291	10.798	1,2,3,4,5,6	0.291	10.798	13.263
Height(m)	599	1,2,4,5	0.099	0.071	1,4,5	0.102	0.072	0.078
BSA(m^2)	599	1,2,3,4,5,6	0.247	0.159	1,2,4,5,6	0.248	0.159	0.189
SBP (mmHg)	158	1,2	0.015	8.860	1,2	0.015	8.860	9.625
DBP (mmHg)	158	1,2	0.026	7.370	1,2	0.026	7.370	7.763
MAP (mmHg)	158	1,2	0.048	7.637	1,2	0.048	7.637	8.430

($R^2 = 0.130, MAE = 6.282$) where linear regression achieved ($R^2 = 0.294, MAE = 5.845$), ($R^2 = 0.247, MAE = 0.159$) and ($R^2 = 0.468, MAE = 4.886$) respectively.

The GNN architecture demonstrated an advantage through its ability to predict multiple target variables simultaneously in a multi-task learning setup, as shown in Table 10.5. This feature allows shared feature learning across targets, which can potentially improve prediction accuracy for interdependent variables while reducing computational overhead.

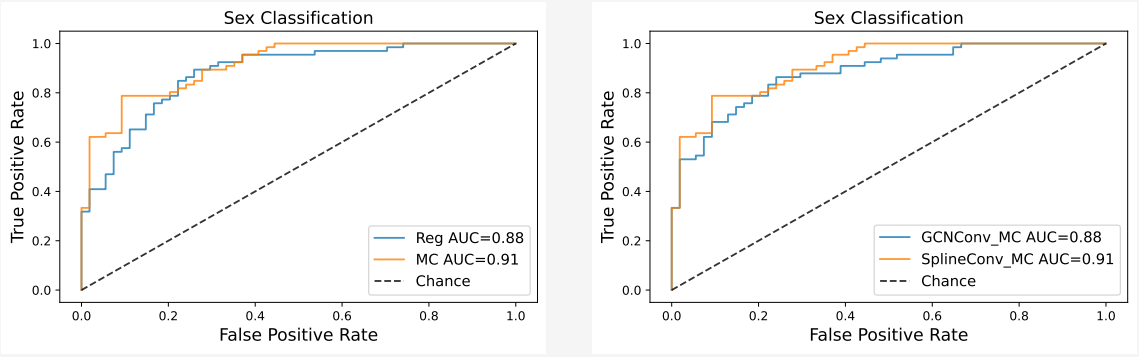
Interestingly, the single-target GNN model better predicted the aortic diameters compared to the multi-target GNN model, except STJ diameter (MAE 1.894 vs 1.862). While multi-target models provide computational efficiency and shared feature learning across interdependent variables, they may sacrifice accuracy for specific targets due to shared capacity constraints or interference between tasks.

Table 10.3: MAE and R^2 for regression models using different combinations of PCA modes after **similarity Umeyama** to predict available target variables

Target	N	Mod Comb (Lowest MAE)	R^2	MAE	Mod Comb (Highest R^2)	R^2	MAE	Median MAE
Age (years)	599	1, 2, 3, 4, 5, 6	0.237	5.904	1, 2, 4, 5, 6	0.238	5.917	6.532
LVEDV(mL)	583	1, 2, 3, 4, 6	0.073	22.869	1, 2, 3, 4, 6	0.073	22.869	24.472
LVESV(mL)	583	1, 3, 4, 5	0.066	12.466	1, 3, 5, 6	0.066	12.513	13.014
LVSV(mL)	583	1, 2, 3, 4	0.040	13.981	1, 2, 3, 4, 6	0.055	13.984	14.819
LVEF(%)	582	1, 2, 5	0.025	4.810	1, 2, 5	0.025	4.810	4.892
LVM(g)	390	1, 2, 3, 5, 6	0.119	21.989	1, 2, 3	0.125	22.028	25.134
HR(bpm)	372	1, 2, 3	0.030	9.005	1, 2, 3	0.030	9.005	9.334
VAC	162	1, 4	0.062	0.551	1, 4, 5, 6	0.088	0.558	0.582
PWV(m/s)	163	2, 3, 5	-0.020	0.839	1, 2, 5	-0.005	0.839	0.787
Mean Flow(mL/s)	318	1, 2, 3	0.025	16.904	1, 2, 3	0.025	16.904	17.390
Annulus dia.(mm)	163	1, 2, 3, 6	0.258	1.889	1, 2, 6	0.258	1.955	2.350
Asc length(mm)	186	1, 2, 4	0.401	5.168	1, 2, 4, 6	0.409	5.178	6.885
Mid sinus dia(mm)	186	1, 2, 4, 6	0.293	2.336	1, 2, 3, 6	0.301	2.342	2.771
Mid sinus height(mm)	186	2, 6	-0.011	2.548	3, 6	-0.009	2.567	2.537
STJ diameter(mm)	186	1, 2, 3, 6	0.320	1.921	1, 2, 6	0.329	1.926	2.447
Sinus height(mm)	186	2, 5, 6	-0.019	3.260	2, 4, 6	-0.004	3.285	3.280
mid Asc dia(mm)	186	1, 2, 6	0.276	1.895	1, 2, 6	0.276	1.895	2.207
end Asc dia(mm)	186	1, 2, 3, 4, 6	0.096	2.538	1, 2, 4, 6	0.103	2.542	2.688
Desc dia(mm)	186	1, 2, 3, 4, 5, 6	0.469	1.463	1, 2, 3, 4, 5, 6	0.469	1.463	2.149
LCS bulge (mm)	186	1, 2, 3, 6	0.280	1.572	1, 2, 3, 6	0.280	1.572	1.910
NCS bulge (mm)	186	2, 5, 6	0.144	1.786	1, 2, 3, 6	0.158	1.807	1.949
RCS bulge (mm)	186	1, 2, 3, 4, 6	0.190	1.466	1, 2, 3, 4, 6	0.190	1.466	1.633
LCS vol (mL)	186	1, 2, 3, 4, 6	0.165	1.170	1, 2, 3, 4, 6	0.165	1.170	1.312
RCS vol (mL)	186	1, 2, 3, 4, 6	0.140	1365.507	1, 2, 3, 4, 6	0.140	1.365	1.453
NCS vol (mL)	186	1, 2, 3, 4, 6	0.126	1.269	1, 2, 3, 4, 6	0.126	1.269	1.375
Weight(Kg)	599	1, 2, 3, 4, 5, 6	0.258	10.710	1, 2, 3, 5, 6	0.261	10.879	13.254
Height(m)	599	1, 2, 3, 5	0.075	0.072	1, 3, 5	0.076	0.072	0.078
BSA(m^2)	599	1, 2, 3, 4, 5, 6	0.226	0.157	1, 2, 3, 5, 6	0.238	0.159	0.190
SBP (mmHg)	158	1, 2	0.048	8.852	1, 6	0.058	8.968	9.566
DBP (mmHg)	158	1, 2	0.024	7.316	1, 2	0.024	7.316	7.671
MAP (mmHg)	158	1, 2	0.063	7.552	1, 2	0.063	7.552	8.576

For size-related parameters like annulus diameter ($R2 = 0.369, MAE = 1.793$), STJ diameter ($R2 = 0.436, MAE = 1.862$) and mid-ascending diameter ($R2 = 0.381, MAE = 1.808$), GNNs outperformed linear regression models, showcasing their ability to capture non-linear relationships, although the improvement over linear regression is modest. The multi-target GNN model also matched linear regression performance for mid-sinus diameter and aortic root size-related parameters. However, the multi-target GNN model underperformed linear regression significantly when predicting blood pressures: SBP, DBP and MAP.

These findings suggest that while GNNs excel at modelling non-linear relationships in size-related parameters, their performance may be constrained by data scarcity or architectural limitations for other target variables.



(a) Effect of mesh processing

(b) Effect of convolutional layer

Figure 10.6: Sex classification performance: (a) using SplineConv for post-processed registered meshes (Reg) vs meshes produced by Marching Cubes (MC) (b) using the meshes that came out of MC with GCNConv vs SplineConv

Table 10.4: Performance of single-target GNN models

Target Variable	**N**	**GNN**		**Linear Regression**	
		R^2	MAE	R^2	MAE
Age (years)	599	-0.276	6.974	0.294	**5.845**
LVM (g)	390	0.186	**21.252**	0.160	22.165
BSA (m^2)	599	0.195	0.163	0.247	**0.159**
Annulus diameter (mm)	163	0.458	**1.698**	0.264	1.861
Mid sinus diameter (mm)	186	0.430	**2.099**	0.277	2.305
STJ diameter (mm)	186	0.393	**1.894**	0.347	1.900
mid Asc. diameter (mm)	186	0.428	**1.684**	0.247	1.885
Desc. diameter (mm)	186	0.568	**1.337**	0.446	1.500
Asc length (mm)	186	0.130	6.282	0.468	**4.886**

Table 10.5: Performance of four multi-target GNN models

Target Variable	N	GNN				Linear Regression	
		R^2	MAE	Norm. R^2	Norm. MAE	R^2	MAE
Annulus Diameter (mm)	163	0.369	1.793	0.398	**1.766**	0.264	1.861
Mid-sinus Diameter (mm)	163	0.331	2.308	0.336	2.309	0.277	2.305
STJ Diameter (mm)	163	0.436	**1.862**	0.353	2.018	0.347	1.900
Mid Asc. Diameter (mm)	163	0.381	1.808	0.437	**1.734**	0.247	1.885
Desc. Diameter (mm)	163	0.349	1.630	0.406	1.540	0.446	**1.500**
Asc. Length (mm)	163	0.266	5.776	0.282	5.662	0.468	**4.886**
SBP (mmHg)	158	-0.380	11.196	-0.449	11.547	0.015	**8.860**
DBP (mmHg)	158	-0.171	9.265	-0.195	9.426	0.026	**7.370**
MAP (mmHg)	158	-0.370	9.959	-0.455	10.309	0.048	**7.637**
LCC bulge (mm)	186	0.284	1.611	0.274	1.635	0.256	**1.592**
NCS bulge (mm)	186	0.270	**1.699**	0.239	1.778	0.129	1.826
RCS bulge (mm)	186	0.197	**1.428**	0.108	1.496	0.168	**1.428**
LCS vol (mL)	186	0.033	1.248	0.010	1.267	0.142	**1.188**
RCS vol (mL)	186	0.044	**1.357**	0.048	1.367	0.139	1.358
NCS vol (mL)	186	0.028	1.268	0.007	1.287	0.159	**1.254**

10.5 Conclusion

This study explored the relationships between aortic shape, characterised by PCA modes, and various clinical parameters using both linear regression models and GNNs. While linear regression models demonstrated strong performance for certain target variables, particularly those with linear relationships to PCA modes, GNNs showed promise in capturing non-linear dependencies and predicting geometric parameters such as aortic diameters at different levels and left ventricular mass. However, GNN performance varied across different targets, with notable underperformance for variables like age, BSA and length of the ascending aorta compared to linear regression, which highlights the importance of selecting the appropriate modelling approach based on the target variable and the underlying relationships. A limitation of this evaluation is that comparisons were confined to a linear regression baseline. While this established a performance floor, it did not assess GNN performance against other non-linear models, such as Random Forest or Gradient Boosting, which might better balance predictive accuracy and computational efficiency. Such comparisons would clarify whether observed improvements arise from the model's non-linearity or the graph-based representation of aortic shape. Nonetheless, linear regression was selected due to its interpretability, widespread use in clinical literature [23, 154, 170], and its role as a minimal benchmark for evaluating the added complexity of GNNs.

One key finding was that single-target GNN models outperformed their multi-target counterparts for specific parameters like aortic diameters. This suggests that while multi-target models offer computational efficiency and the advantage of shared feature learning across interdependent variables, they may sacrifice accuracy for certain targets due to task interference or shared capacity constraints. However, the ability to predict multiple target variables from aortic shape data has significant potential for clinical application, such as early detection of aortic aneurysms and personalised treatment planning.

Additionally, the preprocessing of mesh data played a critical role in model performance. Raw meshes produced by marching cubes yielded better results than registered meshes, highlighting the importance of preserving geometric features during preprocessing.

Overall, this study underscores the potential of GNNs for modelling complex relationships between aortic shape and clinical parameters. However, further refinements in GNN architecture, data augmentation strategies, and preprocessing methods are necessary to fully leverage their capabilities. Future work should focus on validating these findings on larger and more diverse datasets, exploring hybrid models that combine the strengths of linear regression and GNNs, and optimising multi-target architectures to improve prediction accuracy across all target variables.

Chapter 11

Conclusions

This thesis has explored the application of advanced machine learning and geometric deep learning techniques to medical imaging, with a particular focus on shape analysis, segmentation, and morphological characterisation of the aorta. By addressing challenges in automated segmentation, mesh processing, and statistical shape analysis, we have developed robust pipelines for extracting and analysing anatomical structures from medical images. In this chapter, we summarise the main conclusions, discuss the limitations of our work, and outline directions for future research.

11.1 Main conclusions

In **Chapter 4**, the use of Fast Point Feature Histogram (FPFH) as node features significantly improved model performance and facilitated generalisation to out-of-distribution data. Since FPFH features were pose-invariant, they eliminated the need for mesh alignment, simplifying the preprocessing pipeline. Both SplineCNN and GCNConv were viable options for convolutional layers, yielding comparable performance. Additionally, data augmentation was essential in GNNs, particularly when the training dataset was limited.

Chapter 5 emphasised the importance of model inspection for understanding the performance characteristics of GNNs. Feature inspection helped clarify what drove the final prediction, particularly distinguishing between the contributions of the GCN submodel and the classification head. This was crucial when feature embeddings were reused for new tasks. Analysing feature embeddings also revealed biases related to data sources and the presence of non-discriminative features in submodels. The findings indicated that relying solely on test accuracy for model selection was insufficient.

In **Chapter 6**, the fusion of imaging and shape features was explored. Combining ResNet-extracted image embeddings with GNN-derived shape embeddings improved predictive accuracy for Alzheimer's Disease (AD) classification and brain age regression. Additionally, evaluating AD classification models using ROC curves rather than AUC alone provided deeper insights into performance differences at clinically relevant classification thresholds.

Chapter 7 introduced a calibration-aware, multi-level explainability framework to address the limitations of existing GNN explainability methods. By integrating global substructure analysis with node-specific insights, the framework provided a more comprehensive understanding of model decisions, bridging the gap between granular and systemic interpretability.

In **Chapter 8**, the effectiveness of transfer learning in automated aorta segmentation was demonstrated through fine-tuning pre-trained models. The strategic use of pre-trained features and task-specific tuning yielded robust performance while balancing computational efficiency and resource allocation.

Chapter 9 presented a comprehensive pipeline for aortic mesh processing, ensuring accurate vertex correspondence across meshes. The process included iso-surface extraction, post-processing, and registration. A deep learning-based approach for mesh registration outperformed traditional methods, demonstrating superior accuracy in establishing vertex correspondence.

Finally, **Chapter 10** focused on aortic shape characterisation. Principal Component Analysis (PCA) quantified healthy aortic shape variability and identified dominant modes of variation. GNNs were efficient in predicting multiple target variables within a single forward pass, albeit with a slight performance trade-off compared to predicting each target separately. Linear regression using PCA modes yielded comparable performance to GNNs, though an exhaustive

approach was necessary to identify the best-performing PCA mode combinations. Strong correlations were found between aortic shape and parameters such as age, left ventricular (LV) mass, body surface area (BSA), weight, systolic blood pressure (SBP), and mean arterial pressure (MAP). We also found that mesh post-processing and registration slightly degraded GNN performance.

11.2 Limitations

While this thesis has demonstrated the potential of advanced machine learning and geometric deep learning techniques for medical imaging, several limitations must be acknowledged to contextualise the findings and guide future research.

While AUC served as a robust metric for comparing model architectures (Chapters 4 and 5), clinical translation requires optimising sensitivity and specificity at task-specific thresholds. This was partially addressed in later chapters (e.g., TPR@FPR in Chapter 6, model calibration in Chapter 7), but future work should involve clinicians to define acceptable trade-offs for real-world deployment, particularly for AD classification and aortic risk stratification.

One significant limitation is the availability and diversity of datasets. While representative, the datasets used in this study were limited in size and scope. Specifically, aortic shape analysis relied on a relatively small dataset that may not fully capture population-level variability or account for differences across imaging protocols. Larger and more diverse datasets are essential to validate the generalizability of the proposed methods and ensure their applicability to broader clinical settings.

The data augmentation techniques employed in this work were relatively simplistic, focusing primarily on perturbing mesh node positions. While these techniques improved model robustness somewhat, they did not leverage more advanced augmentation methods that could further enhance performance. For instance, recent studies have explored adaptive graph adjustments or edge removal strategies that dynamically modify the graph structure based on predictions or random sampling. Incorporating such advanced augmentation techniques could improve model robustness and generalisation, particularly in scenarios with limited training data.

Another limitation pertains to the specificity of findings derived from certain tasks. For example, conclusions about GNN explainability were based exclusively on sex classification from brain shapes. Similarly, aortic shape characterisation focused solely on healthy volunteers. These task-specific insights may not generalise to other prediction tasks or pathological conditions. Validation across diverse tasks and datasets is necessary to ensure broader applicability.

The fusion methods employed for combining imaging and shape features were relatively basic, relying on feature concatenation. While this approach demonstrated promising results, it does not fully exploit the potential of advanced fusion techniques such as attention mechanisms or adaptive fusion strategies. These methods could enable more nuanced integration of multi-modal data representations, potentially yielding further improvements in predictive accuracy.

Computational requirements posed another challenge. Deep learning-based segmentation and registration models, while effective, demand substantial computational resources. For instance, full training of nn-UNet provided slight performance gains but required significantly higher computational power compared to fine-tuned models like TotalSegmentator or Med-

SAM2. This limits the accessibility of these methods in resource-constrained settings, highlighting the need for more efficient algorithms that balance performance with computational cost.

Specific limitations related to aortic segmentation also emerged during this study. The E-HVols dataset included consistent ground-truth annotations that constrained the descending aorta to the diaphragm level. This discrepancy influenced pre-trained model performance, as demonstrated by TotalSegmentator's reliance on fine-tuning for optimal results. Additionally, MedSAM2 exhibited tendencies for oversegmentation due to its prompt-dependent nature; however, postprocessing modules such as connected-component labelling were not implemented due to inference constraints tied to unavailable ground truth prompts.

Mesh processing methods also introduced limitations due to representation choices. The Marching Cubes algorithm was used for iso-surface extraction, followed by mesh post-processing steps that reduced vertex counts to 1000 per mesh. This may have resulted in information loss, particularly in capturing fine details of mesh curvature.

Finally, this work treated anatomical shapes as static meshes without accounting for dynamic physiological processes such as cardiac motion. For example, time-resolved cardiac MRI data could provide insights into structure-function relationships critical for understanding functional impairments in aortic analysis. Similarly, neuroimaging meshes derived from single-timepoint MRIs may fail to capture patterns of brain shrinkage over time caused by neurodegenerative diseases.

In summary, while this thesis has made significant contributions to advancing geometric deep learning applications in medical imaging, addressing these limitations will be crucial for future research to improve generalizability, robustness, computational efficiency, and clinical relevance.

11.3 Future work

Future work should evaluate the proposed methods on larger and more diverse datasets, such as ADNI [95] for Alzheimer's disease classification and multi-centre cohorts for aortic shape analysis or the German National Cohort [43]. This will help validate their generalizability and robustness across different populations and imaging protocols. Additionally, exploring advanced data augmentation techniques for graph neural networks (GNNs), such as edge removal or adaptive graph adjustments, could improve model performance, particularly in scenarios with limited training data.

Investigating more sophisticated fusion methods, such as attention mechanisms or adaptive fusion, could enhance the integration of imaging and shape features, leading to better predictive performance. Expanding the fusion model to include additional modalities, such as functional MRI or diffusion tensor imaging, could provide a more comprehensive understanding of brain structure and function, further improving predictive accuracy. Additionally, research into GNN explainability methods is needed to ensure the reliability and clinical relevance of these techniques. Our calibration-aware framework should be extended to other tasks such as disease classification and validated with clinical experts to confirm its clinical applicability.

Extending the aortic shape characterisation pipeline to pathological conditions, such as aortic aneurysms or dissections, could provide insights into disease mechanisms and improve diagnostic accuracy. Furthermore, developing more computationally efficient algorithms for segmentation, registration, and shape analysis could enhance the accessibility of these methods in clinical settings with limited resources.

This thesis has demonstrated the potential of advanced machine learning and geometric deep learning techniques for medical imaging and shape analysis. By addressing challenges in segmentation, mesh processing, and statistical shape analysis, we have developed robust pipelines for extracting and analysing anatomical structures. While our findings are promising, there is significant scope for further research to address limitations and explore new directions. We hope this work contributes to the growing body of knowledge in medical imaging and inspires future advancements in the field.

References

[1] Aguib, Y., Allouba, M., Afify, A., et al. “The Egyptian Collaborative Cardiac Genomics (ECCO-GEN) Project: defining a healthy volunteer cohort”. In: *NPJ genomic medicine* 5.1 (2020), pp. 1–8.

[2] Ahmed, E., Saint, A., Shabayek, A. E. R., et al. “A survey on deep learning advances on different 3D data representations”. In: *arXiv preprint arXiv:1808.01462* (2018).

[3] Alharbi, H., Juanatas, R. A., Al Hejaili, A., et al. “Spectral graph convolutional neural network for Alzheimer’s disease diagnosis and multi-disease categorization from functional brain changes in magnetic resonance images”. In: *Frontiers in Neuroinformatics* 18 (2024), p. 1495571.

[4] Amenta, N. “The crust algorithm for 3 D surface reconstruction”. In: *Annual Symposium on Computational Geometry: Proceedings of the fifteenth annual symposium on Computational geometry*. Vol. 13. 16. 1999, pp. 423–424.

[5] Amenta, N., Choi, S., Dey, T. K., et al. “A simple algorithm for homeomorphic surface reconstruction”. In: *Proceedings of the sixteenth annual symposium on Computational geometry*. 2000, pp. 213–222.

[6] Amenta, N., Choi, S., and Kolluri, R. K. “The power crust”. In: *Proceedings of the sixth ACM symposium on Solid modeling and applications*. 2001, pp. 249–266.

[7] El-Assy, A., Amer, H. M., Ibrahim, H., et al. “A novel CNN architecture for accurate early detection and classification of Alzheimer’s disease using MRI data”. In: *Scientific Reports* 14.1 (2024), p. 3463.

[8] Ayed, I. B., Wang, M., Miles, B., et al. “TRIC: trust region for invariant compactness and its application to abdominal aorta segmentation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2014, pp. 381–388.

[9] Azzolin, S., Longa, A., Barbiero, P., et al. “Global explainability of gnns via logic combination of learned concepts”. In: *arXiv preprint arXiv:2210.07147* (2022).

[10] Badki, A., Gallo, O., Kautz, J., et al. “Meshlet priors for 3d mesh reconstruction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 2849–2858.

[11] Bashyam, V. M., Erus, G., Doshi, J., et al. “MRI signatures of brain age and disease over the lifespan based on a deep brain network and 14 468 individuals worldwide”. In: *Brain* 143.7 (2020), pp. 2312–2324.

[12] Battaglia, P. W., Hamrick, J. B., Bapst, V., et al. “Relational inductive biases, deep learning, and graph networks”. In: *CoRR* abs/1806.01261 (2018). arXiv: 1806.01261.

[13] Bernardini, F., Mittleman, J., Rushmeier, H., et al. “The ball-pivoting algorithm for surface reconstruction”. In: *IEEE transactions on visualization and computer graphics* 5.4 (1999), pp. 349–359.

[14] Bessadok, A., Mahjoub, M. A., and Rekik, I. “Graph neural networks in network neuroscience”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.5 (2022), pp. 5833–5848.

[15] Bhatla, P. and Nielsen, J. C. “Cardiovascular magnetic resonance as an alternate method for serial evaluation of proximal aorta: comparison with echocardiography”. In: *Echocardiography* 30.6 (2013), pp. 713–718.

[16] Bintsi, K.-M., Baltatzis, V., Potamias, R. A., et al. “Multimodal brain age estimation using interpretable adaptive population-graph learning”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2023, pp. 195–204.

[17] Boissonnat, J.-D. and Geiger, B. “Three-dimensional reconstruction of complex shapes based on the Delaunay triangulation”. In: *Biomedical image processing and biomedical visualization*. Vol. 1905. SPIE. 1993, pp. 964–975.

[18] Bookstein, F. L. “Thin-plate splines and the atlas problem for biomedical images”. In: *Information Processing in Medical Imaging: 12th International Conference, IPMI’91 Wye, UK, July 7–12, 1991 Proceedings 12*. Springer. 1991, pp. 326–342.

[19] Bronstein, M. M., Bruna, J., LeCun, Y., et al. “Geometric deep learning: going beyond euclidean data”. In: *IEEE Signal Processing Magazine* 34.4 (2017), pp. 18–42.

[20] Bronstein, M. M., Bruna, J., Cohen, T., et al. “Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges”. In: *CoRR* abs/2104.13478 (2021). arXiv: 2104.13478.

[21] Bruse, J. L., McLeod, K., Biglino, G., et al. “A statistical shape modelling framework to extract 3D shape biomarkers from medical imaging data: assessing arch morphology of repaired coarctation of the aorta”. In: *BMC medical imaging* 16.1 (2016), pp. 1–19.

[22] Buchake, V., Muthal, A., Saudagar, R., et al. “A neurodegenerative disorder-Alzheimer disease: A treatise”. In: *Research Journal of Pharmacology and Pharmacodynamics* 2.4 (2010), pp. 268–273.

[23] Campens, L., Demulier, L., De Groote, K., et al. “Reference values for echocardiographic assessment of the diameter of the aortic root and ascending aorta spanning all age categories”. In: *The American journal of cardiology* 114.6 (2014), pp. 914–920.

[24] Cao, W., Yan, Z., He, Z., et al. “A comprehensive survey on geometric deep learning”. In: *IEEE Access* 8 (2020), pp. 35929–35949.

[25] Cao, Z., Huang, Q., and Karthik, R. “3D object classification via spherical projections”. In: *2017 international conference on 3D Vision (3DV)*. IEEE. 2017, pp. 566–574.

[26] Capps, S. B., Elkins, R. C., and Fronk, D. M. “Body surface area as a predictor of aortic and pulmonary valve diameter”. In: *The Journal of thoracic and cardiovascular surgery* 119.5 (2000), pp. 975–982.

[27] Chaari, N., Gharsallaoui, M. A., Akdağ, H. C., et al. “Multigraph classification using learnable integration network with application to gender fingerprinting”. In: *Neural Networks* 151 (2022), pp. 250–263.

[28] Chen, D., Lin, Y., Li, W., et al. “Measuring and Relieving the Over-Smoothing Problem for Graph Neural Networks from the Topological View”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.04 (Apr. 2020), pp. 3438–3445. DOI: 10.1609/aaai.v34i04.5747.

[29] Chen, H., Gomez, C., Huang, C.-M., et al. “Explainable medical imaging AI needs human-centered design: guidelines and evidence from a systematic review”. In: *NPJ digital medicine* 5.1 (2022), p. 156.

[30] Chen, Y. and Medioni, G. “Object modelling by registration of multiple range images”. In: *Image and vision computing* 10.3 (1992), pp. 145–155.

[31] Chen, Z., Yang, F., and Tao, W. “Detarnet: Decoupling translation and rotation by siamese network for point cloud registration”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 1. 2022, pp. 401–409.

[32] Choy, C., Gwak, J., and Savarese, S. “4d spatio-temporal convnets: Minkowski convolutional neural networks”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 3075–3084.

[33] Chu, Z., Wan, Y., Li, Q., et al. “Graph neural networks for vulnerability detection: A counterfactual explanation”. In: *Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis*. 2024, pp. 389–401.

[34] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., et al. “3D U-Net: learning dense volumetric segmentation from sparse annotation”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*. Springer. 2016, pp. 424–432.

[35] Cohen, J. *Statistical power analysis for the behavioral sciences*. 2nd ed. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988.

[36] Cole, J. H. and Franke, K. “Predicting age using neuroimaging: innovative brain ageing biomarkers”. In: *Trends in neurosciences* 40.12 (2017), pp. 681–690.

[37] D’Antonoli, T. A., Berger, L. K., Indrakanti, A. K., et al. “TotalSegmentator MRI: Sequence-Independent Segmentation of 59 Anatomical Structures in MR images”. In: *arXiv preprint arXiv:2405.19492* (2024).

[38] Dai, A. and Nießner, M. “Scan2mesh: From unstructured range scans to 3d meshes”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 5574–5583.

[39] Dai, E. and Wang, S. “Towards self-explainable graph neural network”. In: *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2021, pp. 302–311.

[40] Daroya, R., Atienza, R., and Cajote, R. “REIN: Flexible mesh generation from point clouds”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2020, pp. 352–353.

[41] Dartora, C., Marseglia, A., Mårtensson, G., et al. “A deep learning model for brain age prediction using minimally preprocessed T1w images as input”. In: *Frontiers in Aging Neuroscience* 15 (2024), p. 1303036.

[42] Dash, S., Shakyawar, S. K., Sharma, M., et al. “Big data in healthcare: management, analysis and future prospects”. In: *Journal of Big Data* 6.1 (2019), pp. 1–25.

[43] de, G. N. C. (C. geschaeftsstelle@ nationale-kohorte. “The German National Cohort: aims, study design and organization”. In: *European journal of epidemiology* 29.5 (2014), pp. 371–382.

[44] Deák, Z., Grimm, J. M., Mueck, F., et al. “Endoleak and in-stent thrombus detection with CT angiography in a thoracic aortic aneurysm phantom at different tube energies using filtered back projection and iterative algorithms”. In: *Radiology* 271.2 (2014), pp. 574–584.

[45] Deng, H., Birdal, T., and Ilic, S. “Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 602–618.

[46] Dey, T. K. and Goswami, S. “Tight cocone: a water-tight surface reconstructor”. In: *Proceedings of the eighth ACM symposium on Solid modeling and applications*. 2003, pp. 127–134.

[47] Ding, K., Xu, Z., Tong, H., et al. “Data augmentation for deep graph learning: A survey”. In: *arXiv preprint arXiv:2202.08235* (2022).

[48] Dinov, I. D. “Linear and Nonlinear Dimensionality Reduction”. In: *Data Science and Predictive Analytics: Biomedical and Health Applications using R*. Cham: Springer International Publishing, 2023, pp. 215–274. ISBN: 978-3-031-17483-4. DOI: 10.1007/978-3-031-17483-4_4.

[49] Duval, A. and Malliaros, F. D. “Graphsvx: Shapley value explanations for graph neural networks”. In: *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part II 21*. Springer. 2021, pp. 302–318.

[50] Dyverfeldt, P., Bissell, M., Barker, A. J., et al. “4D flow cardiovascular magnetic resonance consensus statement”. In: *Journal of Cardiovascular Magnetic Resonance* 17.1 (2015), pp. 1–19.

[51] Edelsbrunner, H. and Mücke, E. P. “Three-dimensional alpha shapes”. In: *ACM Transactions On Graphics (TOG)* 13.1 (1994), pp. 43–72.

[52] Erbel, R. and Eggebrecht, H. “Aortic dimensions and the risk of dissection”. In: *Heart* 92.1 (2006), pp. 137–142.

[53] Erler, P., Guerrero, P., Ohrhallinger, S., et al. “Points2surf learning implicit surfaces from point clouds”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 108–124.

[54] Farshian, A., Götz, M., Cavallaro, G., et al. “Deep-Learning-Based 3-D Surface Reconstruction—A Survey”. In: *Proceedings of the IEEE* (2023).

[55] Faul, F., Erdfelder, E., Lang, A.-G., et al. “G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences”. In: *Behavior research methods* 39.2 (2007), pp. 175–191.

[56] Ferguson, M. K. *Thoracic surgery atlas*. Elsevier Health Sciences, 2007.

[57] Fey, M., Lenssen, J. E., Weichert, F., et al. “Splinecnn: Fast geometric deep learning with continuous b-spline kernels”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 869–877.

[58] Fortune, S. “Voronoi diagrams and Delaunay triangulations”. In: *Handbook of discrete and computational geometry*. Chapman and Hall/CRC, 2017, pp. 705–721.

[59] Franke, K., Ziegler, G., Klöppel, S., et al. “Estimating the age of healthy subjects from T1-weighted MRI scans using kernel methods: exploring the influence of various parameters”. In: *Neuroimage* 50.3 (2010), pp. 883–892.

[60] Gao, L., Yang, J., Wu, T., et al. “SDM-NET: Deep generative network for structured deformable mesh”. In: *ACM Transactions on Graphics (TOG)* 38.6 (2019), pp. 1–15.

[61] Gao, S. and Lima, D. “A review of the application of deep learning in the detection of Alzheimer’s disease”. In: *International Journal of Cognitive Computing in Engineering* 3 (2022), pp. 1–8.

[62] Gaser, C., Franke, K., Klöppel, S., et al. “BrainAGE in mild cognitive impaired patients: predicting the conversion to Alzheimer’s disease”. In: *PloS one* 8.6 (2013), e67346.

[63] Gatehouse, P. D., Keegan, J., Crowe, L. A., et al. “Applications of phase-contrast flow and velocity imaging in cardiovascular MRI”. In: *European radiology* 15.10 (2005), pp. 2172–2184.

[64] Gelaude, F., Vander Sloten, J., and Lauwers, B. “Accuracy assessment of CT-based outer surface femur meshes”. In: *Computer Aided Surgery* 13.4 (2008), pp. 188–199.

[65] Gilmer, J., Schoenholz, S. S., Riley, P. F., et al. “Neural message passing for quantum chemistry”. In: *International conference on machine learning*. PMLR. 2017, pp. 1263–1272.

[66] Girard, N. and Tarabalka, Y. “End-to-end learning of polygons for remote sensing image classification”. In: *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2018, pp. 2083–2086.

[67] Gkioxari, G., Malik, J., and Johnson, J. “Mesh r-cnn”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 9785–9795.

[68] Glocker, B., Jones, C., Roschewitz, M., et al. “Risk of bias in chest radiography deep learning foundation models”. In: *Radiology: Artificial Intelligence* 5.6 (2023), e230060.

[69] Gong, S., Chen, L., Bronstein, M., et al. “Spiralnet++: A fast and highly efficient mesh convolution operator”. In: *Proceedings of the IEEE/CVF international conference on computer vision workshops*. 2019, pp. 0–0.

[70] Gori, M., Monfardini, G., and Scarselli, F. “A new model for learning in graph domains”. In: *Proceedings. 2005 IEEE international joint conference on neural networks, 2005.* Vol. 2. IEEE. 2005, pp. 729–734.

[71] Graham, B., Engelcke, M., and Van Der Maaten, L. “3d semantic segmentation with submanifold sparse convolutional networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 9224–9232.

[72] Grande-Allen, K. J., Cochran, R. P., Reinhall, P. G., et al. “Re-creation of sinuses is important for sparing the aortic valve: a finite element study”. In: *The Journal of thoracic and cardiovascular surgery* 119.4 (2000), pp. 753–763.

[73] Groueix, T., Fisher, M., Kim, V. G., et al. “A papier-mâché approach to learning 3d surface generation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 216–224.

[74] Gundelwein, L., Ramm, H., Goubergrits, L., et al. “3D Shape analysis for coarctation of the Aorta”. In: *International workshop on shape in medical imaging*. Springer. 2018, pp. 73–77.

[75] Guo, C., Pleiss, G., Sun, Y., et al. “On calibration of modern neural networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 1321–1330.

[76] Guo, Y., Wang, H., Hu, Q., et al. “Deep learning for 3D point clouds: A survey. arXiv 2019”. In: *arXiv preprint arXiv:1912.12033* ().

[77] Guo, Y., Wang, H., Hu, Q., et al. “Deep learning for 3d point clouds: A survey”. In: *IEEE transactions on pattern analysis and machine intelligence* (2020).

[78] El-Hamamsy, I. “The living aortic valve”. In: (2010).

[79] Hanik, M., Demirtaş, M. A., Gharsallaoui, M. A., et al. “Predicting cognitive scores with graph neural networks through sample selection learning”. In: *Brain imaging and behavior* 16.3 (2022), pp. 1123–1138.

[80] Hanocka, R., Hertz, A., Fish, N., et al. “Meshcnn: a network with an edge”. In: *ACM Transactions on Graphics (ToG)* 38.4 (2019), pp. 1–12.

[81] Hanocka, R., Metzer, G., Giryes, R., et al. “Point2mesh: A self-prior for deformable meshes”. In: *arXiv preprint arXiv:2005.11084* (2020).

[82] Heinrich, M. P. and Oster, J. “MRI whole heart segmentation using discrete nonlinear registration and fast non-local fusion”. In: *International Workshop on Statistical Atlases and Computational Models of the Heart*. Springer. 2017, pp. 233–241.

[83] Hepp, T., Fischer, M., Winkelmann, M. T., et al. “Fully Automated Segmentation and Shape Analysis of the Thoracic Aorta in Non–contrast-enhanced Magnetic Resonance Images of the German National Cohort Study”. In: *Journal of thoracic imaging* 35.6 (2020), pp. 389–398.

[84] Hong, G., Wang, Z., Han, T., et al. “Spatiotemporal multi-graph convolutional network for taxi demand prediction”. In: *2021 11th International Conference on Information Science and Technology (ICIST)*. IEEE. 2021, pp. 242–250.

[85] Hu, Q., Yang, B., Xie, L., et al. “Randla-net: Efficient semantic segmentation of large-scale point clouds”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11108–11117.

[86] Hu, S.-M., Liu, Z.-N., Guo, M.-H., et al. “Subdivision-based mesh convolution networks”. In: *ACM Transactions on Graphics (TOG)* 41.3 (2022), pp. 1–16.

[87] Huang, Q., Yamada, M., Tian, Y., et al. “Graphlime: Local interpretable model explanations for graph neural networks”. In: *IEEE Transactions on Knowledge and Data Engineering* 35.7 (2022), pp. 6968–6972.

[88] Huang, X., Li, S., Zuo, Y., et al. “Unsupervised point cloud registration by learning unified gaussian mixture models”. In: *IEEE Robotics and Automation Letters* 7.3 (2022), pp. 7028–7035.

[89] Huang, Z., Li, K., Wang, S., et al. “SES: Bridging the Gap Between Explainability and Prediction of Graph Neural Networks”. In: *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE. 2024, pp. 2945–2958.

[90] Ibrahim, A. M., Roshdy, M., Latif, N., et al. “Structural and Functional Characterization of the Aorta in Hypertrophic Obstructive Cardiomyopathy”. In: *Circulation: Heart Failure* 18.2 (2025), e012384.

[91] Iglesias, J. E., Liu, C.-Y., Thompson, P. M., et al. “Robust brain extraction across datasets and comparison with publicly available methods”. In: *IEEE Transactions on Medical Imaging* 30.9 (2011), pp. 1617–1634.

[92] Isensee, F., Jaeger, P. F., Kohl, S. A., et al. “nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation”. In: *Nature methods* 18.2 (2021), pp. 203–211.

[93] Isensee, F., Petersen, J., Klein, A., et al. “nnu-net: Self-adapting framework for u-net-based medical image segmentation”. In: *arXiv preprint arXiv:1809.10486* (2018).

[94] Isensee, F., Wald, T., Ulrich, C., et al. “nnu-net revisited: A call for rigorous validation in 3d medical image segmentation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2024, pp. 488–498.

[95] Jack Jr, C. R., Bernstein, M. A., Fox, N. C., et al. “The Alzheimer’s disease neuroimaging initiative (ADNI): MRI methods”. In: *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine* 27.4 (2008), pp. 685–691.

[96] Jack Jr, C. R., Petersen, R. C., Xu, Y. C., et al. “Medial temporal atrophy on MRI in normal aging and very mild Alzheimer’s disease”. In: *Neurology* 49.3 (1997), pp. 786–794.

[97] Jacob, P., Zablocki, É., Ben-Younes, H., et al. “STEEX: steering counterfactual explanations with semantics”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 387–403.

[98] Jiang, H., Shen, Y., Xie, J., et al. “Sampling network guided cross-entropy method for unsupervised point cloud registration”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 6128–6137.

[99] Jónsson, B. A., Bjornsdottir, G., Thorgeirsson, T. E., et al. “Brain age prediction using deep learning uncovers associated sequence variants”. In: *Nature communications* 10.1 (2019), p. 5409.

[100] Kaiser, T., Kellenberger, C. J., Albisetti, M., et al. “Normal values for aortic diameters in children and adolescents–assessment in vivo by contrast-enhanced CMR-angiography”. In: *Journal of Cardiovascular Magnetic Resonance* 10 (2008), pp. 1–8.

[101] Katayama, S., Umetani, N., Sugiura, S., et al. “The sinus of Valsalva relieves abnormal stress on aortic valve leaflets by facilitating smooth closure”. In: *The Journal of thoracic and cardiovascular surgery* 136.6 (2008), pp. 1528–1535.

[102] Kazhdan, M., Bolitho, M., and Hoppe, H. “Poisson surface reconstruction”. In: *Proceedings of the fourth Eurographics symposium on Geometry processing*. Vol. 7. 4. 2006.

[103] Kazi, A., Shekarforoush, S., Arvind Krishna, S., et al. “Graph convolution based attention model for personalized disease prediction”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2019, pp. 122–130.

[104] Kazmi, I. K., You, L., and Zhang, J. J. “A survey of 2D and 3D shape descriptors”. In: *2013 10th International Conference Computer Graphics, Imaging and Visualization*. IEEE. 2013, pp. 1–10.

[105] Kelly, C. J., Karthikesalingam, A., Suleyman, M., et al. “Key challenges for delivering clinical impact with artificial intelligence”. In: *BMC medicine* 17 (2019), pp. 1–9.

[106] Khemani, B., Patil, S., Kotecha, K., et al. “A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions”. In: *Journal of Big Data* 11.1 (2024), p. 18.

[107] Kim, B.-H., Ye, J. C., and Kim, J.-J. “Learning dynamic graph representation of brain connectome with spatio-temporal attention”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 4314–4327.

[108] Kingma, D. P. and Ba, J. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[109] Kipf, T. N. and Welling, M. “Semi-supervised classification with graph convolutional networks”. In: *arXiv preprint arXiv:1609.02907* (2016).

[110] Kirillov, A., Mintun, E., Ravi, N., et al. “Segment anything”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 4015–4026.

[111] Kull, M., Perello Nieto, M., Kängsepp, M., et al. “Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration”. In: *NeurIPS* 32 (2019).

[112] LaMontagne, P. J., Benzinger, T. L., Morris, J. C., et al. “OASIS-3: longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and Alzheimer disease”. In: *MedRxiv* (2019).

[113] Lansac, E., Lim, H., Shomura, Y., et al. “Aortic and pulmonary root: are their dynamics similar?” In: *European Journal of Cardio-thoracic Surgery* 21.2 (2002), pp. 268–275.

[114] Lee, S.-L., Aguib, H., Chapron, J., et al. “Spatial orientation and morphology of the pulmonary artery: relevance to optimising design and positioning of a continuous pressure monitoring device”. In: *Journal of Cardiovascular Translational Research* 9.3 (2016), pp. 239–248.

[115] Lee, M., Lee, J., and Chang, J.-H. “Ensemble of jointly trained deep neural network-based acoustic models for reverberant speech recognition”. In: *Digital Signal Processing* 85 (2019), pp. 1–9.

[116] Li, C. *OpenAI’s GPT-3 Language model: A technical overview*. Sept. 2020.

[117] Li, D., Lu, C., Chen, Z., et al. “Graph Neural Networks in Point Clouds: A Survey”. In: *Remote Sensing* 16.14 (2024), p. 2518.

[118] Li, J., Cai, D., and He, X. “Learning graph-level representation for drug discovery”. In: *arXiv preprint arXiv:1709.03741* (2017).

[119] Li, M., Jiang, Y., Zhang, Y., et al. “Medical image analysis using deep learning algorithms”. In: *Frontiers in public health* 11 (2023), p. 1273253.

[120] Li, Y., Bu, R., Sun, M., et al. “Pointcnn: Convolution on x-transformed points”. In: *Advances in neural information processing systems* 31 (2018).

[121] Li, Y., Daho, M. E. H., Conze, P.-H., et al. “A review of deep learning-based information fusion techniques for multimodal medical image classification”. In: *Computers in Biology and Medicine* (2024), p. 108635.

[122] Liao, Y., Donne, S., and Geiger, A. “Deep marching cubes: Learning explicit surface representations”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2916–2925.

[123] Liem, F., Varoquaux, G., Kynast, J., et al. “Predicting brain-age from multimodal imaging data captures cognitive impairment”. In: *Neuroimage* 148 (2017), pp. 179–188.

[124] Lin, K., Wang, L., and Liu, Z. “End-to-end human pose and mesh reconstruction with transformers”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 1954–1963.

[125] Lin, Y., Yan, Z., Huang, H., et al. “Fpconv: Learning local flattening for point convolution”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 4293–4302.

[126] Litmanovich, D., Bankier, A. A., Cantin, L., et al. “CT and MRI in diseases of the aorta”. In: *American Journal of Roentgenology* 193.4 (2009), pp. 928–940.

[127] Liu, M., Zhang, X., and Su, H. “Meshing point clouds with predicted intrinsic-extrinsic ratio guidance”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*. Springer. 2020, pp. 68–84.

[128] Lorensen, W. E. and Cline, H. E. “Marching cubes: A high resolution 3D surface construction algorithm”. In: *Seminal graphics: pioneering efforts that shaped the field*. 1998, pp. 347–353.

[129] Luan, J., Mao, L., Zhu, Z., et al. “New indicators for systematic assessment of aortic morphology: a narrative review”. In: *Journal of Thoracic Disease* 13.1 (2021). ISSN: 2077-6624.

[130] Luo, D., Cheng, W., Xu, D., et al. “Parameterized explainer for graph neural network”. In: *Advances in neural information processing systems* 33 (2020), pp. 19620–19631.

[131] Luo, N., Shi, W., Yang, Z., et al. “Multimodal fusion of brain imaging data: Methods and applications”. In: *Machine Intelligence Research* 21.1 (2024), pp. 136–152.

[132] Lyu, M., Yang, J., Qi, Z., et al. “Rigid pairwise 3D point cloud registration: a survey”. In: *Pattern Recognition* (2024), p. 110408.

[133] Ma, J., He, Y., Li, F., et al. “Segment Anything in Medical Images”. In: *Nature Communications* 15 (2024), p. 654.

[134] Ma, J., Kim, S., Li, F., et al. “Segment anything in medical images and videos: Benchmark and deployment”. In: *arXiv preprint arXiv:2408.03322* (2024).

[135] Matloff, N. *Statistical regression and classification: from linear models to machine learning*. Chapman and Hall/CRC, 2017.

[136] Mavridis, C., Vagenas, T. P., Economopoulos, T. L., et al. “Automatic Segmentation in 3D CT Images: A Comparative Study of Deep Learning Architectures for the Automatic Segmentation of the Abdominal Aorta”. In: *Electronics* 13.24 (2024), p. 4919.

[137] Menagadevi, M., Devaraj, S., Madian, N., et al. “Machine and deep learning approaches for alzheimer disease detection using magnetic resonance images: An updated review”. In: *Measurement* (2024), p. 114100.

[138] Monti, F., Boscaini, D., Masci, J., et al. “Geometric deep learning on graphs and manifolds using mixture model cnns”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 5115–5124.

[139] Morris, C., Ritzert, M., Fey, M., et al. “Weisfeiler and leman go neural: Higher-order graph neural networks”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 4602–4609.

[140] Morris, J. C. “The clinical dementia rating (cdr): Current version and”. In: *Young* 41 (1991), pp. 1588–1592.

[141] Muksimova, S., Umirzakova, S., Iskhakova, N., et al. “Advanced convolutional neural network with attention mechanism for Alzheimer’s disease classification using MRI”. In: *Computers in Biology and Medicine* 190 (2025), p. 110095.

[142] Myronenko, A. and Song, X. “Point set registration: Coherent point drift”. In: *IEEE transactions on pattern analysis and machine intelligence* 32.12 (2010), pp. 2262–2275.

[143] Nienaber, C. A. and Powell, J. T. “Management of acute aortic syndromes”. In: *European heart journal* 33.1 (2012), pp. 26–35.

[144] Omran, M., Engelbrecht, A., and Salman, A. “An overview of clustering methods, Intelligent Data Analysis”. In: (2007).

[145] Pais, G. D., Ramalingam, S., Govindu, V. M., et al. “3dregnet: A deep neural network for 3d point registration”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 7193–7203.

[146] Parisot, S., Ktena, S. I., Ferrante, E., et al. “Disease prediction using graph convolutional networks: application to autism spectrum disorder and Alzheimer’s disease”. In: *Medical image analysis* 48 (2018), pp. 117–130.

[147] Parker, K. H., Jones, C. J., Dawson, J. R., et al. “What stops the flow of blood from the heart?” In: *Heart and vessels* 4.4 (1988), pp. 241–245.

[148] Paszke, A., Gross, S., Massa, F., et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in neural information processing systems* 32 (2019).

[149] Patenaude, B., Smith, S. M., Kennedy, D. N., et al. “A Bayesian model of shape and appearance for subcortical brain segmentation”. In: *Neuroimage* 56.3 (2011), pp. 907–922.

[150] Paul, J. “BESL and Neil MCKAY, A methode for registration of 3d shapes”. In: *IEEE Transactions on pattern analysis and machine intelligence* 14.2 (1992), pp. 239–256.

[151] Pauly, M. *Point primitives for interactive modeling and processing of 3D geometry*. Hartung-Gorre Süderdithmarschen, Germany, 2003.

[152] Piegl, L. and Tiller, W. *The NURBS book*. Springer Science & Business Media, 1996.

[153] Pinto-Coelho, L. “How artificial intelligence is shaping medical imaging technology: a survey of innovations and applications”. In: *Bioengineering* 10.12 (2023), p. 1435.

[154] Poutanen, T., Tikanoja, T., Sairanen, H., et al. “Normal aortic dimensions and flow in 168 children and young adults”. In: *Clinical physiology and functional imaging* 23.4 (2003), pp. 224–229.

[155] Qi, C. R., Su, H., Mo, K., et al. “Pointnet: Deep learning on point sets for 3d classification and segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 652–660.

[156] Qi, C. R., Yi, L., Su, H., et al. “Pointnet++: Deep hierarchical feature learning on point sets in a metric space”. In: *Advances in neural information processing systems* 30 (2017).

[157] Qian, G., Li, Y., Peng, H., et al. “Pointnext: Revisiting pointnet++ with improved training and scaling strategies”. In: *Advances in neural information processing systems* 35 (2022), pp. 23192–23204.

[158] Qiu, Y., Liu, W., Wang, J., et al. “PAGE: Parametric Generative Explainer for Graph Neural Network”. In: *arXiv preprint arXiv:2408.14042* (2024).

[159] Rachel Draelos MD, P. *Measuring Performance: AUC (AUROC)*. https://glassboxmedicine.com/2019/02/23/measuring-performance-auc-auroc/. [Online; accessed 21-January-2025]. 2019.

[160] Rani, A., Ortiz-Arroyo, D., and Durdevic, P. “Advancements in point cloud-based 3D defect classification and segmentation for industrial systems: A comprehensive survey”. In: *Information Fusion* (2024), p. 102575.

[161] Rathore, S., Habes, M., Iftikhar, M. A., et al. “A review on neuroimaging-based classification studies and associated feature extraction methods for Alzheimer’s disease and its prodromal stages”. In: *NeuroImage* 155 (2017), pp. 530–548.

[162] Ravi, N., Gabeur, V., Hu, Y.-T., et al. “Sam 2: Segment anything in images and videos”. In: *arXiv preprint arXiv:2408.00714* (2024).

[163] Ravi, N., Reizenstein, J., Novotny, D., et al. “Accelerating 3D Deep Learning with PyTorch3D”. In: *arXiv:2007.08501* (2020).

[164] Ravi, N., Reizenstein, J., Novotny, D., et al. “Accelerating 3d deep learning with pytorch3d”. In: *arXiv preprint arXiv:2007.08501* (2020).

[165] Rayed, M. E., Islam, S. S., Niha, S. I., et al. “Deep learning for medical image segmentation: State-of-the-art advancements and challenges”. In: *Informatics in Medicine Unlocked* (2024), p. 101504.

[166] Raz, N., Lindenberger, U., Rodrigue, K. M., et al. “Regional brain changes in aging healthy adults: general trends, individual differences and modifiers”. In: *Cerebral cortex* 15.11 (2005), pp. 1676–1689.

[167] Robertson, T., Wright, F., and Dykstra, R. *Order Restricted Statistical Inference*. John Wiley & Sons, 1988.

[168] Robinson, R. D. “Reliable Machine Learning for Medical Imaging Data through Automated Quality Control and Data Harmonization”. In: (July 2020).

[169] Rodero, C., Strocchi, M., Marciniak, M., et al. “Linking statistical shape models and simulated function in the healthy adult human heart”. In: *PLoS computational biology* 17.4 (2021), e1008851.

[170] Roman, M. J., Devereux, R. B., Kramer-Fox, R., et al. “Two-dimensional echocardiographic aortic root dimensions in normal children and adults”. In: *The American journal of cardiology* 64.8 (1989), pp. 507–512.

[171] Rong, Y., Huang, W., Xu, T., et al. “Dropedge: Towards deep graph convolutional networks on node classification”. In: *arXiv preprint arXiv:1907.10903* (2019).

[172] Ronneberger, O., Fischer, P., and Brox, T. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer. 2015, pp. 234–241.

[173] Ruigrok, A. N., Salimi-Khorshidi, G., Lai, M.-C., et al. “A meta-analysis of sex differences in human brain structure”. In: *Neuroscience & Biobehavioral Reviews* 39 (2014), pp. 34–50.

[174] Rusu, R. B., Blodow, N., and Beetz, M. “Fast point feature histograms (FPFH) for 3D registration”. In: *2009 IEEE international conference on robotics and automation*. IEEE. 2009, pp. 3212–3217.

[175] Rusu, R. B., Marton, Z. C., Blodow, N., et al. “Learning informative point classes for the acquisition of object model maps”. In: *2008 10th International Conference on Control, Automation, Robotics and Vision*. IEEE. 2008, pp. 643–650.

[176] Salti, S., Tombari, F., and Di Stefano, L. “SHOT: Unique signatures of histograms for surface and texture description”. In: *Computer Vision and Image Understanding* 125 (2014), pp. 251–264.

[177] Sarasua, I., Pölsterl, S., and Wachinger, C. “Hippocampal representations for deep learning on Alzheimer’s disease”. In: *Scientific reports* 12.1 (2022), pp. 1–13.

[178] Schlichtkrull, M. S., De Cao, N., and Titov, I. “Interpreting graph neural networks for NLP with differentiable edge masking”. In: *arXiv preprint arXiv:2010.00577* (2020).

[179] Sedghi Gamechi, Z., Bons, L. R., Giordano, M., et al. “Automated 3D segmentation and diameter measurement of the thoracic aorta on non-contrast enhanced CT”. In: *European radiology* 29 (2019), pp. 4613–4623.

[180] Seo, H., Badiei Khuzani, M., Vasudevan, V., et al. “Machine learning techniques for biomedical image segmentation: an overview of technical aspects and introduction to state-of-art applications”. In: *Medical physics* 47.5 (2020), e148–e167.

[181] Shafto, M. A., Tyler, L. K., Dixon, M., et al. “The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) study protocol: a cross-sectional, lifespan, multidisciplinary examination of healthy cognitive ageing”. In: *BMC Neurology* 14.1 (2014), p. 204.

[182] Shan, C., Shen, Y., Zhang, Y., et al. “Reinforcement learning enhanced explainer for graph neural networks”. In: *NeurIPS* 34 (2021), pp. 22523–22533.

[183] Sharp, N. and Ovsjanikov, M. “Pointtrinet: Learned triangulation of 3d point sets”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*. Springer. 2020, pp. 762–778.

[184] Shehata, N. “A personalised approach towards aortic root and ascending aorta surgery”. MRes thesis. Unpublished thesis. 2017.

[185] Shehata, N., Bain, W., and Glocker, B. “A Comparative Study of Graph Neural Networks for Shape Classification in Neuroimaging”. In: *Proceedings of the First International Workshop on Geometric Deep Learning in Medical Image Analysis*. Ed. by Bekkers, E., Wolterink, J. M., and Aviles-Rivero, A. Vol. 194. Proceedings of Machine Learning Research. PMLR, 18 Nov 2022, pp. 160–171.

[186] Shehata, N., Piçarra, C., Kazi, A., et al. “The Importance of Model Inspection for Better Understanding Performance Characteristics of Graph Neural Networks”. In: *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*. 2024, pp. 1–5. DOI: 10.1109/ISBI56570.2024.10635526.

[187] Shen, Y., Hui, L., Jiang, H., et al. “Reliable inlier evaluation for unsupervised point cloud registration”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 2. 2022, pp. 2198–2206.

[188] Shi, B., Bai, S., Zhou, Z., et al. “Deeppano: Deep panoramic representation for 3-d shape recognition”. In: *IEEE Signal Processing Letters* 22.12 (2015), pp. 2339–2343.

[189] Shlomi, J., Battaglia, P., and Vlimant, J.-R. “Graph neural networks in particle physics”. In: *Machine Learning: Science and Technology* 2.2 (2020), p. 021001.

[190] Shrikumar, A., Greenside, P., and Kundaje, A. “Learning important features through propagating activation differences”. In: *International conference on machine learning*. PMlR. 2017, pp. 3145–3153.

[191] Sinha, A., Bai, J., and Ramani, K. “Deep learning 3D shape surfaces using geometry images”. In: *European Conference on Computer Vision*. Springer. 2016, pp. 223–240.

[192] Strubell, E., Ganesh, A., and McCallum, A. “Energy and policy considerations for deep learning in NLP”. In: *arXiv preprint arXiv:1906.02243* (2019).

[193] Su, H., Maji, S., Kalogerakis, E., et al. “Multi-view convolutional neural networks for 3d shape recognition”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 945–953.

[194] Suk, H.-I., Lee, S.-W., Shen, D., et al. “Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis”. In: *NeuroImage* 101 (2014), pp. 569–582.

[195] Sundararajan, M., Taly, A., and Yan, Q. “Axiomatic attribution for deep networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 3319–3328.

[196] Swarnkar, S. K., Guru, A., Chhabra, G. S., et al. “Deep learning techniques for medical image segmentation & classification”. In: *International journal of health sciences* 6.S10 (2022), pp. 408–421.

[197] Tanveer, M., Ganaie, M., Beheshti, I., et al. “Deep learning for brain age estimation: A systematic review”. In: *Information Fusion* 96 (2023), pp. 130–143.

[198] Tatarchenko, M., Dosovitskiy, A., and Brox, T. “Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2088–2096.

[199] Taylor, J. R., Williams, N., Cusack, R., et al. “The Cambridge Centre for Ageing and Neuroscience (Cam-CAN) data repository: structural and functional MRI, MEG, and cognitive data from a cross-sectional adult lifespan sample”. In: *NeuroImage* 144 (2017), pp. 262–269. DOI: https://doi.org/10.1016/j.neuroimage.2015.09.018.

[200] Thamsen, B., Yevtushenko, P., Gundelwein, L., et al. “Unsupervised Learning and Statistical Shape Modeling of the Morphometry and Hemodynamics of Coarctation of the Aorta”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2020, pp. 776–785.

[201] Thomas, H., Qi, C. R., Deschaud, J.-E., et al. “Kpconv: Flexible and deformable convolution for point clouds”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6411–6420.

[202] Tops, L. F., Wood, D. A., Delgado, V., et al. “Noninvasive evaluation of the aortic root with multislice computed tomography: implications for transcatheter aortic valve replacement”. In: *JACC: Cardiovascular Imaging* 1.3 (2008), pp. 321–330.

[203] Torii, R., El-Hamamsy, I., Donya, M., et al. “Integrated morphologic and functional assessment of the aortic root after different tissue valve root replacement procedures”. In: *The Journal of thoracic and cardiovascular surgery* 143.6 (2012), pp. 1422–1428.

[204] Tustison, N. J., Avants, B. B., Cook, P. A., et al. “N4ITK: improved N3 bias correction”. In: *IEEE Transactions on Medical Imaging* 29.6 (2010), pp. 1310–1320.

[205] Ueda, T., Chin, A., Petrovitch, I., et al. “A pictorial review of acute aortic syndrome: discriminating and overlapping features as revealed by ECG-gated multidetector-row CT angiography”. In: *Insights into imaging* 3.6 (2012), pp. 561–571.

[206] Umeyama, S. “Least-squares estimation of transformation parameters between two point patterns”. In: *IEEE Transactions on pattern analysis and machine intelligence* 13.4 (1991), pp. 376–380.

[207] Uyguroğlu, F., Toygar, Ö., and Demirel, H. “CNN-based Alzheimer’s disease classification using fusion of multiple 3D angular orientations”. In: *Signal, Image and Video Processing* 18.3 (2024), pp. 2743–2751.

[208] Verma, N., Boyer, E., and Verbeek, J. “Feastnet: Feature-steered graph convolutions for 3d shape analysis”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 2598–2606.

[209] Vinodkumar, P. K., Karabulut, D., Avots, E., et al. “A survey on deep learning based segmentation, detection and classification for 3D point clouds”. In: *Entropy* 25.4 (2023), p. 635.

[210] Voges, I., Jerosch-Herold, M., Hedderich, J., et al. “Normal values of aortic dimensions, distensibility, and pulse wave velocity in children and young adults: a cross-sectional study”. In: *Journal of Cardiovascular Magnetic Resonance* 14.1 (2012), p. 41.

[211] Vriz, O., Aboyans, V., D’Andrea, A., et al. “Normal values of aortic root dimensions in healthy adults”. In: *The American journal of cardiology* 114.6 (2014), pp. 921–927.

[212] Vu, M. and Thai, M. T. “Pgm-explainer: Probabilistic graphical model explanations for graph neural networks”. In: *Advances in neural information processing systems* 33 (2020), pp. 12225–12235.

[213] Wang, H., Shen, Y., Wang, S., et al. “Ensemble of 3D densely connected convolutional network for diagnosis of mild cognitive impairment and Alzheimer’s disease”. In: *Neurocomputing* 333 (2019), pp. 145–156.

[214] Wang, W., Chen, J., Zhang, Y., et al. “A multi-graph convolutional network framework for tourist flow prediction”. In: *ACM Transactions on Internet Technology (TOIT)* 21.4 (2021), pp. 1–13.

[215] Wang, X., Wu, Y., Zhang, A., et al. “Causal screening to interpret graph neural networks”. In: (2021).

[216] Wang, X. and Shen, H.-W. “Gnninterpreter: A probabilistic generative model-level explanation for graph neural networks”. In: *arXiv preprint arXiv:2209.07924* (2022).

[217] Wang, Y. and Solomon, J. “Intrinsic and extrinsic operators for shape analysis”. In: *Handbook of Numerical Analysis*. Vol. 20. Elsevier, 2019, pp. 41–115.

[218] Wang, Y., Sun, Y., Liu, Z., et al. “Dynamic graph cnn for learning on point clouds”. In: *ACM Transactions on Graphics (tog)* 38.5 (2019), pp. 1–12.

[219] Wiggers, K. *OpenAI’s massive GPT-3 model is impressive, but size isn’t everything*. June 2020.

[220] Wolak, A., Gransar, H., Thomson, L. E., et al. “Aortic Size Assessment by Noncontrast Cardiac Computed Tomography: Normal Limits by Age, Gender, and Body Surface Area”. In: *Journal of Cardiovascular Computed Tomography* 2.1 (2008), pp. 18–27.

[221] Wright, F., Warncke, M., Sinn, M., et al. “Assessment of aortic diameter in Marfan patients: intraindividual comparison of 3D-Dixon and 2D-SSFP magnetic resonance imaging”. In: *European Radiology* 33.3 (2023), pp. 1687–1697.

[222] Wu, Z., Pan, S., Chen, F., et al. “A comprehensive survey on graph neural networks”. In: *IEEE transactions on neural networks and learning systems* 32.1 (2020), pp. 4–24.

[223] Xie, Y., Katariya, S., Tang, X., et al. “Task-agnostic graph explanations”. In: *NeurIPS* 35 (2022), pp. 12027–12039.

[224] Xu, H., Ye, N., Liu, G., et al. “FINet: Dual branches feature interaction for partial-to-partial point cloud registration”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 3. 2022, pp. 2848–2856.

[225] Xu, K., Hu, W., Leskovec, J., et al. “How powerful are graph neural networks?” In: *arXiv preprint arXiv:1810.00826* (2018).

[226] Yacoub, M. “Valve-conserving operation for aortic root aneurysm or dissection”. In: *Operative Techniques in Cardiac and Thoracic Surgery* 1.1 (1996), pp. 57–67.

[227] Yacoub, M. H., Nagy, M., Hosny, H., et al. “Right ventricular structure and function after novel repair of common arterial trunk”. In: *International Journal of Cardiology Congenital Heart Disease* 10 (2022), p. 100416.

[228] Yang, Z., Liu, W., Gan, H., et al. “Alzheimer’s disease classification based on brain region-to-sample graph convolutional network”. In: *Biomedical Signal Processing and Control* 96 (2024), p. 106589.

[229] Ying, Z., Bourgeois, D., You, J., et al. “Gnnexplainer: Generating explanations for graph neural networks”. In: *Advances in neural information processing systems* 32 (2019).

[230] Yuan, H., Tang, J., Hu, X., et al. “Xgnn: Towards model-level explanations of graph neural networks”. In: *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 2020, pp. 430–438.

[231] Yuan, H., Yu, H., Gui, S., et al. “Explainability in graph neural networks: A taxonomic survey”. In: *IEEE transactions on pattern analysis and machine intelligence* 45.5 (2022), pp. 5782–5799.

[232] Yuan, H., Yu, H., Wang, J., et al. “On explainability of graph neural networks via subgraph explorations”. In: *International conference on machine learning*. PMLR. 2021, pp. 12241–12252.

[233] Yuksel, C. “Sample elimination for generating Poisson disk sample sets”. In: *Computer Graphics Forum*. Vol. 34. 2. Wiley Online Library. 2015, pp. 25–32.

[234] Zeng, H.-M., Han, H.-B., Zhang, Q.-F., et al. “Application of modern neuroimaging technology in the diagnosis and study of Alzheimer’s disease”. In: *Neural regeneration research* 16.1 (2021), pp. 73–79.

[235] Zhang, C., Yuan, G., and Tao, W. “DMNet: Delaunay Meshing Network for 3D Shape Representation”. In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society. 2023, pp. 14372–14382.

[236] Zhang, D., Wang, Y., Zhou, L., et al. “Multimodal classification of Alzheimer’s disease and mild cognitive impairment”. In: *Neuroimage* 55.3 (2011), pp. 856–867.

[237] Zhang, L., Fonseca, M. J. da, Ferreira, A., et al. “Survey on 3D shape descriptors”. In: *FundaÃgao para a Cincia ea Tecnologia, Lisboa, Portugal, Tech. Rep. Technical Report, DecorAR (FCT POSC/EIA/59938/2004)* 3 (2007).

[238] Zhang, Y.-X., Gui, J., Cong, X., et al. “A Comprehensive Survey and Taxonomy on Point Cloud Registration Based on Deep Learning”. In: *arXiv preprint arXiv:2404.13830* (2024).

[239] Zhang, Y. and Rabbat, M. “A graph-cnn for 3d point cloud classification”. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 6279–6283.

[240] Zhang, Y., Defazio, D., and Ramesh, A. “Relex: A model-agnostic relational model explainer”. In: *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 2021, pp. 1042–1049.

[241] Zhang, Z. “Variable selection with stepwise and best subset approaches”. In: *Annals of translational medicine* 4.7 (2016), p. 136.

[242] Zhao, H., Jiang, L., Jia, J., et al. “Point transformer”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16259–16268.

[243] Zhou, H., He, L., Chen, B. Y., et al. “Multi-modal diagnosis of Alzheimer’s disease using interpretable graph convolutional networks”. In: *IEEE Transactions on Medical Imaging* (2024).

[244] Zhou, J., Shen, J., and Xuan, Q. “Data augmentation for graph classification”. In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2020, pp. 2341–2344.

[245] Zhou, J., Cui, G., Hu, S., et al. “Graph neural networks: A review of methods and applications”. In: *AI Open* 1 (2020), pp. 57–81.

[246] Zhou, Q.-Y., Park, J., and Koltun, V. “Fast global registration”. In: *European conference on computer vision*. Springer. 2016, pp. 766–782.

[247] Zhou, Q.-Y., Park, J., and Koltun, V. “Open3D: A modern library for 3D data processing”. In: *arXiv preprint arXiv:1801.09847* (2018).

[248] Zhou, Z., Wang, Q., An, X., et al. “A novel graph neural network method for Alzheimer’s disease classification”. In: *Computers in Biology and Medicine* 180 (2024), p. 108869.

[249] Zhu, H. and Koniusz, P. “Simple spectral graph convolution”. In: *International Conference on Learning Representations*. 2020.

[250] Zhuang, X., Li, L., Payer, C., et al. “Evaluation of algorithms for multi-modality whole heart segmentation: an open-access grand challenge”. In: *Medical image analysis* 58 (2019), p. 101537.

Appendices

The GitHub repository at https://github.com/mlnotebook/thesis_template contains the code used in this work.