



UNIVERSITY OF LEEDS

Graph-based Probabilistic Geometric Deep Learning Framework for Generation of Virtual Anatomical Populations



Soodeh Kalaie

University of Leeds

School of Computing

Submitted in accordance with the requirements for the degree of

Doctor of Philosophy

November, 2023

*To my wonderful family, whose unwavering support and love have been
constant companions throughout my academic journey.*

Intellectual Property Statement

The candidate confirms that the work submitted is her own, except where work which has formed part of jointly authored publications has been included. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others. The contribution of the candidate and the other authors to this work has been explicitly indicated as follows.

As the main author, I took the lead in designing the study and writing the manuscript, which included analyzing and discussing the results. I also contributed to writing the analysis code and performed data processing, statistical analysis, and experiments. The other authors provided valuable input in designing the study, providing data, participating in result discussions, and reviewing the manuscript.

- Chapter 4:
 - 1) S.Kalaie, A.Zakeri, Z.A.Taylor, A.F.Frangi, A.Gooya. "Deep Generative Modeling for Shapes using Graph Convolution Networks and Attention.", *Pattern Recognition*, Under review.
- Chapter 5:
 - 1) S.Kalaie, A.Bulpitt, A.F.Frangi, A.Gooya. "A Geometric Deep Learning Framework for Generation of Virtual Left Ventricles as Graphs." *Medical Imaging with Deep Learning (MIDL)*. 2023.
 - 2) S.Kalaie, A.Bulpitt, A.F.Frangi, A.Gooya. "An End-to-End Deep Learning Generative Framework for Refinable Shape Matching and Generation: Towards Comprehensive Anatomical Shape Analysis and Synthesis.", Submitted.

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper

acknowledgement.

The right of Soodeh Kalaie to be identified as Author of this work has been asserted by her in accordance with the Copyright, Designs and Patents Act 1988.

© 2023The University of Leeds and Soodeh Kalaie.

Acknowledgements

I am deeply grateful to the numerous individuals who have supported me throughout my PhD journey. Their support has been invaluable, with some offering valuable technical advice, while others provided much-needed encouraging words, especially during the challenging times of the Pandemic. Without them, I would not have enjoyed this journey, and this thesis would likely not have been possible.

First, I would like to thank my supervisor, Dr. Ali Gooya, for his technical guidance and support, and for many lessons about the academic world. I also want to thank my second supervisor, Professor Alejandro Frangi, for giving me the opportunity to pursue a PhD degree and work alongside him at CISTIB. I am thankful for his enthusiastic leadership, continuous constructive criticism and friendly advice, all of which helped me become a better scientist. A special thanks goes to my last supervisor, Professor Andy Bulpitt, for his endless professional support during the final year of my research.

Besides my supervisors, I would also like to express my gratitude to my co-authors, for sharing their knowledge and skills with me, and I eagerly look forward to future collaborations with each one of them. To my fellow labmates at CISTIB, thank you for the stimulating discussions over coffee and the cherished memories we have created together.

Last but not least, I would like to thank those who have been a constant source of support. To my family and friends in my home country, who made every return feel as if I had never left, I hope that feeling never fades. To my sister and brother, for our daily random scientific or meme discussions, thank you for being good friends to me. Finally, my deepest gratitude goes to my parents. Their unwavering love and continuous support have been the cornerstone of my life. Thank you for believing in me, often even more than I believed in myself, and for your boundless encouragement.

Abstract

In the field of medical imaging, "shape generation" refers to the computational techniques employed to create accurate and detailed representations of anatomical structures/organs. Shape generation plays a crucial role in medical imaging, profoundly impacting clinical applications and diagnostics. Accurate representation of anatomical structures is essential for disease detection, treatment planning, personalized medicine, and computational modelling. Leveraging machine learning and computational modelling opens avenues for valuable insights through In-Silico Clinical Trials (ISCTs).

In ISCTs, virtual populations of anatomical shapes are vital for evaluating clinical devices. These populations must capture anatomical and physiological variability while remaining plausible to ensure meaningful and reliable results. By generating virtual shape populations, researchers can simulate and assess medical interventions, accelerating the development of improved therapies and devices. However, constructing generative models faces challenges due to the fact that real-world anatomical shapes, derived from different subjects, exhibit varying topological structures and, in general, there is no topological correspondence among shapes from different subjects. The present study focuses on generating representative populations of anatomical shapes to overcome these obstacles.

This thesis aims to address the challenges associated with shape matching and generation by introducing an unsupervised probabilistic deep generative model, applicable to datasets including shape surface mesh data with varying topological structures and the absence of correspondences. The proposed framework leverages graph representations to capture the geometric characteristics of anatomical shapes and incorporates advanced techniques in geometric deep learning. By employing these algorithms, the framework is able to establish a learnable set of vertex-wise correspondences between shapes in the latent space while learning/constructing a population-derived atlas model. Subsequently, the model generates virtual populations of anatomical shapes that closely resemble real-world data. This novel generative framework is designed to handle variable topology in anatomical structures across patients/input shapes and successfully synthesises anatomically

plausible virtual populations with significant variability in shape and diverse topologies. These capabilities expand the potential applications of the approach in computational medicine and make it well-suited for ISCTs. Furthermore, the experimental results on image-derived mesh datasets show the superiority of the proposed approach as the synthesised virtual anatomical shape populations are more plausible (i.e. higher specificity) and capture a greater degree of variability (i.e. generalisation) in shape than those generated by the baseline shape models.

CONTENTS

1	Introduction	1
1.1	Significance of Anatomical Shape Modeling in Medical Imaging	2
1.2	Why Shape Generation Matters?	2
1.3	Virtual Population Modeling	5
1.4	Shape Representation	6
1.5	Thesis Contributions and Overview	7
2	Literature Review	10
2.1	Introduction	11
2.2	Deep Learning on Non-Euclidean Data	11
2.2.1	Graph Construction	13
2.2.2	Graph Convolutional Neural Networks	14
2.3	Generative Shape Modeling	21
2.3.1	Establishing Correspondences	22
2.3.2	Shape Generation	24
2.4	Conclusion	27
3	Baseline Generative Shape Model: Datasets, Evaluation Metrics, and Performance Analysis	29
3.1	Datasets	30
3.1.1	Pre-Alignment of Shapes	31
3.2	Evaluation Metrics	31
3.2.1	Generalisation and Specificity	32
3.2.2	Clinical Relevance	33
3.3	Baseline Generative Shape Model	34

3.3.1	Point-Set Registration: Coherent Point Drift	36
3.3.2	Statistical Shape Models	38
3.4	Experimental Results	40
3.5	Discussion and Conclusion	47
4	A Geometric Deep Learning Framework for Generation of Shape as Graphs Using Graph Convolution and Attention Mechanisms: Leveraging Geometry and Surface Features	50
4.1	Introduction	51
4.2	ASMG Model	52
4.2.1	Attention-based Shape Matching (ASM)	55
4.2.2	Generative Modeling	63
4.3	Results and Discussion	64
4.3.1	Evaluation of GCN-ATT Network Performance	66
4.3.2	Evaluation of Generation	75
4.4	Conclusion and Limitations	84
5	An End-to-End Deep Learning Framework for Refinable Shape Matching and Generation: Enhancing Specificity and Generalisability	87
5.1	Introduction	88
5.2	Atlas-R-ASMG model	89
5.2.1	Refinable Attention-based Shape Matching (R-ASM)	91
5.2.2	Generation Network	93
5.2.3	Unsupervised Loss	93
5.2.4	Atlas Construction	95
5.3	Results and Discussion	96
5.3.1	Evaluation of Shape Matching	96
5.3.2	Evaluation of Generation	105
5.4	Joint Clustering Generative Shape Model	113
5.4.1	Clustering Scheme	115
5.4.2	Generative Modeling	117
5.4.3	Experimental Results and Discussion	118
5.5	Conclusion	122

6	Conclusions	124
6.1	Summary and Achievements	125
6.2	Limitations and Future Research Directions	128
A	Appendix	130
A.1	Shape smoothing using Laplacian over \mathbf{h}_k	131
A.2	Optimising the template shape in a closed form solution	131
	References	132

LIST OF FIGURES

1.1	Conducting in-silico trials of medical imaging devices; Simulation of imaging processes (top) using virtual imaging trials (bottom) [1].	4
2.1	Illustration examples of convolution on (a) Euclidian data and (b) non-Euclidian data [2].	12
2.2	(a) Illustration of a standard CNN, representing the parameters as a set $M = w \times h$ weight matrices. Each weight matrix is associated with a single relative position in the input patch. (b) Feature-Steered graph convolution network [3], where each node (e.g. i) in the input patch is softly associated to each of the M weight matrices based on their features using $q_m(x_i, x_j)$	20
3.1	Visualisation of some shape matching results obtained by registration-based methods described in Section 3.3.1. Some landmarks are highlighted with red arrows. (a) Cyan-coloured meshes present input LV shapes with cardinality 1586, 1539, 1455, 1150 and 1039 respectively from left to right. Grey-coloured shapes are the normalised meshes with a cardinality of 1093. (b) Cyan-coloured meshes present input liver shapes with cardinality 1532, 1338, 1278, 1030 and 1365 respectively from left to right. Grey-coloured shapes are the normalised meshes with a cardinality of 1025.	41

3.2	The Cumulative Explained Variance (CEV) and the shape variation explained by each of the principal components of the feature vector in the space defined by the PCA, in order of importance. (a) The first 13 modes explain 95% of the total LV shape variation. (b) The first 10 modes explain 95% of the total liver shape variation.	42
3.3	Representation of the mean ± 3 standard deviation (SD) of the first five modes of variation in the 3D shape models of left-ventricle.	43
3.4	Representation of the mean ± 3 standard deviation (SD) of the first five modes of variation in the 3D shape models of the liver.	45
3.5	Generalisation and Specificity errors (in [mm]) of the RSMP(Uni) model with different principal components L . The first and second rows illustrate results on LV and liver datasets, respectively. The plots show the performance of the models with L values smaller than 32. In each row, HD , ED and ED^* distances are reported from left to right. For each model, the markers, dotted lines and, arrows indicate the associated average, confidence ellipse, and the direction of the maximum variability (in error values for test and synthetic data) respectively, for training with various numbers of components.	46
4.1	Schematic showing our geometric deep learning framework for shape matching and generation. Different-sized actual shapes first go through the shape matching phase, in which, the network derives a learnable set of correspondences and learns population-derived atlas shape, then flows into the generation network, and finally produces the realistic synthetic 3D shapes.	52
4.2	Overview of the proposed ASMG framework for graph generation from a graph population with different graph structures.	54
4.3	Illustration of the proposed attention-based shape matching procedure which finds the vertex-wise correspondence between two shapes by measuring similarities between the nodal embeddings.	57

4.4	Visualisation and Quantification of the established vertex-to-vertex correspondences by sGCN-ATT for four cases of LV shapes. Examples for illustrating meaningful correspondences (blue lines) between a sparse set of landmark points (pink) on an atlas and subjects; color-coded vertices on the shapes present the value of soft attention map ($\phi_{ji} \in [0, 1]; j \in \{\text{landmark points}\}, i = \{1, \dots, N_k\}$).	67
4.5	Visualisation and Quantification of the established vertex-to-vertex correspondences by sGCN-ATT for four cases of liver shapes. Examples for illustrating meaningful correspondences (blue lines) between a sparse set of landmark points (pink) on an atlas and subjects; color-coded vertices on the shapes present the value of soft attention map ($\phi_{ji} \in [0, 1]; j \in \{\text{landmark points}\}, i = \{1, \dots, N_k\}$).	68
4.6	Qualitative shape matching results using the ASM method (in s/hGCN-ATT settings) on the LV dataset. The atlas shape is shown merely for visualisation purposes (colour code).	69
4.7	Qualitative shape matching results using the ASM method (in s/hGCN-ATT settings) on the liver dataset. The atlas shape is shown merely for visualisation purposes (colour code).	69
4.8	Visualization of the shape regressions and graph Laplacian smoothing. The results are presented for the LV and liver samples. In each shape category, the top and bottom present the regressed (\mathbf{h}_k) and smoothed shapes (\mathbf{h}'_k) obtained using Equation (4.3) and Equation (4.5), respectively.	70
4.9	Examples show the comparison between different shape matching approaches RSM and ASM (s/hGCN-ATT) for deriving correspondence on the LV dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1586, 1539, 1455, 1150$ and 1039 respectively from left to right. Grey-coloured shapes are the normalised meshes g'_k with cardinality $N_\mu = 1093$ obtained from different methods. Notice endocardial, epicardial, LV base, and apex landmarks. Some of the landmarks are shown by red arrows.	72

4.10	Examples show the comparison between different shape matching approaches RSM and ASM (s/hGCN-ATT) for deriving correspondence on the liver dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1532, 1338, 1278, 1030$ and 1365 respectively from left to right. Grey-coloured shapes are the normalised meshes g'_k with cardinality $N_\mu = 1025$ obtained from different methods. Red arrows highlight some landmarks.	73
4.11	Impact of atlas resolution on graph regression on the atlas, and normalising their structures using established vertex-to-vertex correspondences for all population samples of LV and liver datasets. Boxplots with line connecting mean values for each group show the higher resolution atlases outperform the low-resolution ones.	75
4.12	The population-derived atlas reconstructed from the (a) LV and (b) liver datasets in two different settings s/hGCN-ATT.	76
4.13	Generalisation and Specificity errors (in [mm]) of the ASMG model with different values of latent dimension L . The first and second rows illustrate results on LV and liver datasets, respectively. The plots show the performance of the models with L values smaller than 64. In each row, HD , ED and ED^* distances are reported from left to right. For each model, the markers, dotted lines and, arrows indicate the associated average, confidence ellipse, and the direction of the maximum variability (in error values for test and synthetic data) respectively, for training with various latent dimensions. A high-performance model illustrates simultaneous small generalisation and specificity errors.	77
4.14	Generalisation and Specificity errors (in [mm]) of the RSMP and s/hGCN-ATT-VAE models with same values of latent dimension $L = 16$. The first and second rows present the results on LV and liver datasets, respectively. In each row, HD , ED and ED^* distances are reported from left to right. For each model, the markers and dotted lines indicate the associated average and confidence ellipse, and the direction of the maximum variability (in error values for test and synthetic data) respectively. A high-performance model illustrates simultaneous small generalisation and specificity errors.	80

4.15	Violin plots for the distribution of LV and liver volume indices on the actual samples, alongside with those generated using different methods: RSMP, sGCN-ATT-VAE, and hGCN-ATT-VAE. Horizontal lines mark the confidence intervals for the different acceptance criteria.	82
4.16	Examples of virtual (LV/liver) samples generated by the ASMG generator model.	84
5.1	Overview of the Atlas Refinable Attention-based Shape Matching and Generation network (Atlas-R-ASMG) framework for shape generation from different-sized shapes.	89
5.2	Qualitative shape matching results using the R-ASM method on the LV dataset. The atlas shape is shown merely for visualisation purposes (colour code).	97
5.3	Qualitative shape matching results using the R-ASM method on the liver dataset. The atlas shape is shown merely for visualisation purposes (colour code).	97
5.4	Examples show the comparison between different shape matching approaches ASM and R-ASM (in two settings s/hGCN-ATT) for deriving correspondence on the LV dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1586, 1539, 1455, 1150$ and 1039 respectively from left to right. Shapes g'_k are the normalised meshes with cardinality $N_\mu = 1093$ obtained from different methods in the s/hGCN-ATT settings. Notice endocardial, epicardial, LV base, and apex landmarks.	99
5.5	Examples show the comparison between different shape matching approaches ASM and R-ASM (in two settings s/hGCN-ATT) for deriving correspondence on the liver dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1532, 1338, 1278, 1030$ and 1365 respectively from left to right. Shapes g'_k are the normalised meshes with cardinality $N_\mu = 1025$ obtained from different methods in the s/hGCN-ATT settings. Red arrows highlight some landmarks.	100
5.6	The population-derived atlas reconstructed from the (a) LV and (b) liver datasets by Atlas-R-ASMG method in two different settings s/hGCN-ATT.	101

5.7 Scatter plots showing HD (left) and CD (right) results of attention-based shape matching frameworks in $[mm]$, with "sGCN-ATT" setting, on the 200 LV and 28 liver cases test set, comparing the ASM (x -axis) versus refinable model R-ASM (y -axis). The green/red gradients indicate an increase/decrease in performance with refinement. Refinement generally improves HD and CD of attention-based shape matching results. Degradation is observed for some LV/liver cases in terms of HD, whereas improvements in HD can be significant. 103

5.8 Scatter plots showing HD(left) and CD (right) results of attention-based shape matching frameworks in $[mm]$, with "hGCN-ATT" setting, on the 200 LV and 28 liver cases test set, comparing the ASM (x -axis) versus refinable model R-ASM (y -axis). The green/red gradients indicate an increase/decrease in performance with refinement. In the majority of cases, HD and CD of the attention-based shape matching results are improved with refinement. Degradation is observed for some LV/liver cases in terms of HD, whereas improvements in HD can be significant. . 104

5.9 Generalisation ability: Boxplots show the generalisation errors (in $[mm]$) for the different models RSMP, ASM, and Atlas-R-ASMG with different settings s/h, where "s" and "h" refer to sGCN-ATT-VAE and hGCN-ATT-VAE respectively and the models' performance are statistically compared. The first and second columns illustrate results on LV and liver datasets, respectively. In each row, HD , ED , and ED^* distances are reported from top to bottom. 107

5.10 Specification ability: Boxplots show the specificity errors (in $[mm]$) for the different models RSMP, ASM, and Atlas-R-ASMG with different settings s/h, where "s" and "h" refer to sGCN-ATT-VAE and hGCN-ATT-VAE respectively and the models' performance are statistically compared. The first and second columns illustrate results on LV and liver datasets, respectively. In each row, HD , ED , and ED^* distances are reported from top to bottom. 108

5.11 Violin plots for the distribution of LV and liver volume indices on the actual samples, alongside with those generated using different methods: RSMP, ASMG(s/h) and Atlas-R-ASMG(s/h). Horizontal lines mark the confidence intervals for the different acceptance criteria. 109

5.12 Examples of three synthetic LV shapes generated by Atlas-R-ASMG(h) model, with decreasing feasibility of the biomarkers according to the acceptance functions. From left to right: g_1^* is accepted by all of the criteria; g_2^* is only rejected by \mathcal{A}^μ but not by the other acceptance functions; and g_3^* is rejected by \mathcal{A}^μ and \mathcal{A}^M but accepted by \mathcal{A}^r 110

5.13 Examples of three synthetic LV shapes generated by Atlas-R-ASMG(h) model, with decreasing feasibility of the biomarkers according to the acceptance functions. From left to right: g_1^* and g_2^* are accepted by all of the criteria; g_3^* is only rejected by \mathcal{A}^r but not by the other acceptance functions. 110

5.14 Examples of virtual (LV/liver) samples generated by the Atlas-R-ASMG generator model. 112

5.15 Graphical model representation of (a) the Atlas-R-ASMG generative model, and (b) joint clustering generative model (mAtlas-R-ASMG). The observed variables are shown with shaded circles and hyper-parameters are indicated by solid small circles/on the arrows. Other variables and parameters are shown with white circles. Please refer to the text for a detailed explanation. 115

5.16 Examples of clustering results on liver datasets, using the Multi-atlas clustering method with $M = 5$ clusters, for establishing shape matching. Arrows indicate the direction of metric improvement. Bold numbers are the best for a given metric/parameter. 120

5.17 Generalisation and Specificity errors (in [mm]) of the mAtlas-R-ASMG model with different values of cluster m . Three distance measures HD , ED and ED^* report the model specificity and generalisation. Boxplots with the line connecting mean values for each group show multi-atlases outperform the single-atlas approaches. 121

LIST OF TABLES

3.1	Statistical description of the biomarkers measured on LV and liver datasets.	35
3.2	Shape matching quality assessment for RSM method using two distance metrics HD and CD (mean \pm std) in $[mm]$.	42
3.3	Clinical acceptance rates \mathcal{A} [in %] achieved by RSMP(Uni) generative model for LV and liver volumes.	47
4.1	Commonly used notations.	53
4.2	The detailed information on the generative model ASMG for two different settings, where matrices \mathbf{C} and \mathbf{n} denote vertices locations and vertices normals respectively, \oplus means a concatenation of matrices.	64
4.3	The structure of network Ψ , where N denotes the cardinality of input shape, $d_z = 128$ and $d_x = 3$ or 6 for s/hGCN settings.	65
4.4	Shape Matching Quality: comparison between different methods using two distance metrics HD and CD (mean \pm std) in $[mm]$. Bold values show a significant difference between the methods with a p-value < 0.001 using the statistical paired t-test.	73
4.5	Generalisation ability: comparison between generative models in terms of generalisation ability using three distance metrics HD , ED and ED^* (mean \pm std) in $[mm]$. Bold values present their performance are significantly better than those of the RSMP model, with a p-value < 0.001 .	78
4.6	Specification ability: comparison between generative models in terms of specificity using three distance metrics HD , ED and ED^* (mean \pm std) in $[mm]$. Bold results present their performance is significantly better than those of the RSMP model, with a p-value < 0.001 .	79

4.7	Clinical acceptance rates \mathcal{A} [in %] achieved by different generative models for LV and liver volumes.	83
5.1	Shape Matching Quality: comparison between different methods; RSM, ASM and R-ASM using two distance metrics HD and CD (mean \pm std) in [mm]. Bold numbers are the best and the second best with the best also underlined, for a given metric. Statistically significant (p-value < 0.01) improvement of the best model over a given model is indicated by superscript *.	101
5.2	Clinical acceptance rates \mathcal{A} [in %] achieved by different generative models for LV and liver volumes.	111
5.3	Shape matching quality in mAtlas-R-ASM model with different number of atlases (i.e. M) on liver shapes. Values show the performance of the model using two distance metrics HD and CD (mean \pm std) in [mm]. Bold values show a significant difference between the methods with a p-value < 0.001 using the statistical paired t-test.	118

Abbreviations

2D	Two-dimensional
3D	Three-dimensional
AI	Artificial intelligence
Atlas-R-ASMG	Atlas Refinable Attention-based Shape Matching and Generation model
ASM	Attention-based Shape Matching
ASMG	Attention-based Shape Matching and Generation model
CMR	Cardiac Magnetic Resonance
CPD	Coherent point drift
CT	Computed Tomography
CAD	Computer-Assisted Diagnosis
CNN	Convolutional Neural Network
CEV	Cumulative Explained Variance
DSC	Dice Similarity Coefficient
DCNN	Diffusion-Convolutional Neural Network
ELBO	Evidence Lower Bound
EM	Expectation-Maximisation
FeaStNet	Feature-Steered Network
GMM	Gaussian Mixture Model
GAN	Generative Adversarial Network
GCNN	Geodesic Convolutional Neural Network
GCN	Graph Convolutional Network
GNN	Graph Neural Network
ISCT	In-Silico Clinical Trial
IoU	Intersection over Union
ICP	Iterative Closest Point

KL	Kullback-Leibler
LV	Left Ventricle
MWA	Mean Weighted by Angle
MWE	Mean Weighted Equally
MoNet	Mixture Model Network
mAtlas-R-ASMG	Multi-Atlas Refinable Attention-based Shape Matching Generative model
NN4G	Neural Network For Graphs
PDM	Point Distribution Model
PCA	Principal Component Analysis
pdf	probability density function
ReLU	Rectified Linear Unit
R-ASM	Refinable Attention-based Shape Matching
RSM	Registration-based Shape Matching
RSMP	Registration-based Shape Matching with a PCA-based gen- erative model
PEMP (Uni)	RSMP using Uniform sampling
Spectral GCN	Spectral-based graph convolutional neural network
SSM	Statistical Shape Model
TCIA	The Cancer Imaging Archive
VAE	Variational AutoEncoder

CHAPTER 1

Introduction

1.1 Significance of Anatomical Shape Modeling in Medical Imaging

Medical imaging techniques play a pivotal role in the diagnosis and management of various diseases and conditions. Among the crucial factors contributing to the efficacy and accuracy of medical imaging is the accurate representation of shape and anatomy. Therefore representation of shape and anatomy within medical images provides critical insights into human health.

Modelling of anatomical shapes provides three-dimensional (3D) shape representation of the inner body structures and organs, making them widely used tools for diagnosis, treatment, surgical planning, and medical training [4–7]. In the clinic, the physical interaction with models facilitates learning anatomy and how different structures interact spatially in the body. The use of anatomical models in simulation-based training reduces the risks involved in surgical interventions, resulting in a better patient experience and reduced healthcare costs. Such reconstructions can be also used for Computer-Assisted Diagnosis (CAD), radiation therapy planning, prosthesis milling, and volumetric measurements. Over the last few decades, numerous shape analysis techniques have been developed for modelling the human anatomy from medical images [8]. These methods have become a mainstay in medical image analysis, not only because they can provide priors for segmentation, but because they can also quantify shape changes between subjects and populations [9, 10]. Thus, a significant focus of medical imaging research is on representing the organ structures.

1.2 Why Shape Generation Matters?

In light of recent advances in machine learning, it is useful to develop a data-driven and learning-based shape analysis framework that can utilize large amounts of data and incorporate prior knowledge. Recent advances in neural networks have revolutionized many fields of medical data analysis.

Artificial intelligence (AI) models are critical technologies for the infrastructure of any modern healthcare system. These AI models are often trained using a population of real patient data to assist with the computer-aided diagnosis [11], risk stratification [12, 13], and prediction of the disease trajectories [14, 15]. However, patient privacy concerns are a significant obstacle which hinders healthcare providers from sharing their

repository of patient data for reliable training of the AI models [16, 17]. One possible remedy is to use artificial generative models capable of generating synthetic datasets resembling the actual data distribution [18].

Moreover, in recent years, the growing complexity and diversity of medical devices and technologies have posed significant challenges in evaluating and optimising their design and clinical applications. While clinical trials are ideal for such evaluations, they are not always practical due to various constraints, such as ethical limitations, costs, time requirements, difficulties in recruiting enough subjects, or a lack of ground truth (knowledge of the patient’s exact anatomy and condition). In-Silico Clinical Trials (ISCTs) are prime examples of applications that can significantly benefit from synthetic data and offer an alternative way to evaluate medical devices virtually [1].

In ISCTs, computational models are used to create virtual cohorts of data for assessing the safety and efficacy of new drugs and the performance of medical devices [19, 20]. ISCTs, therefore, offer the potential to investigate the performance of medical devices across a broader range of patient characteristics than is feasible for a real clinical trial. These trials could help refine, reduce, and partially replace in vivo clinical trials. In contrast to actual patients, computational models are characterized by their exact anatomy, providing a "gold standard" or "ground truth" to evaluate the efficiency of medical devices quantitatively. For instance, in medical imaging, ISCTs replace the human subject with virtual digital phantoms, the imaging system with virtual simulated scanners, and the clinical interpretation with virtual interpretations. Figure 1.1 illustrates an ISCT framework for medical imaging. ISCTs permit researchers to answer fundamental questions quickly and cost-effectively using precise controls and known ground truth, which are only possible in a virtual environment. As a result of these virtual trials, new and existing imaging technologies are objectively evaluated in terms of their utility and diagnostic accuracy, while minimising any potential risks (e.g., radiation dose).

More specifically, in-silico trials of medical devices are an advanced technology that has the potential to revolutionize the design, development, or even regulatory evaluation of the devices used to treat pathologies. The availability of virtual populations is essential for conducting large-scale in-silico trials of medical devices. The lack of clinical data poses a barrier to setting up population-based in silico trials, thus the development of generative models of anatomy, representative of real-world patient populations is a

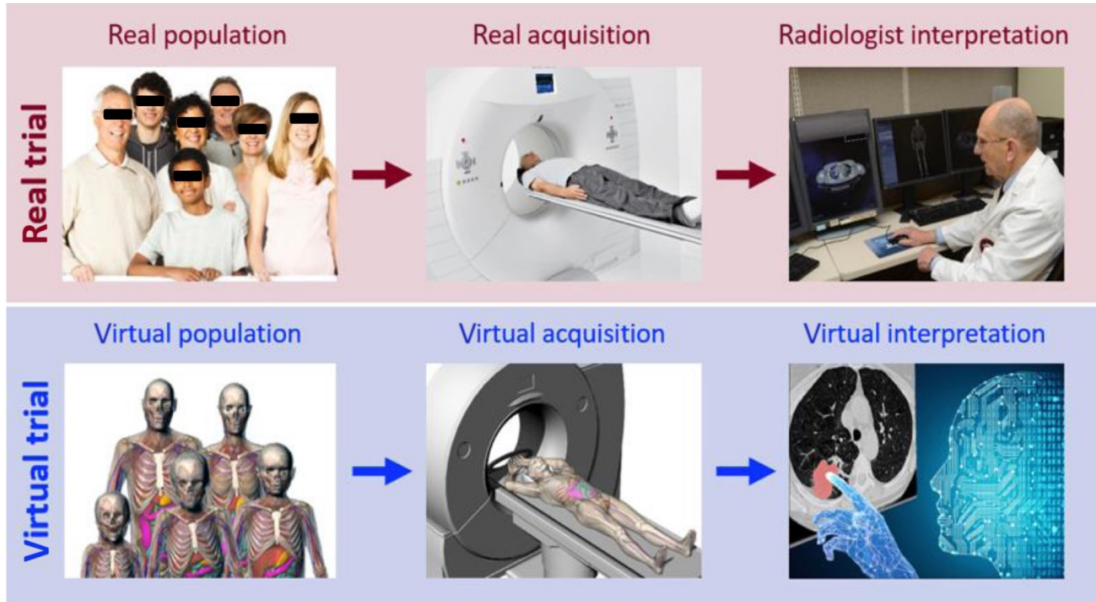


Figure 1.1: Conducting in-silico trials of medical imaging devices; Simulation of imaging processes (top) using virtual imaging trials (bottom) [1].

prerequisite for ISCTs. In order to overcome this limitation, generative models can be used to create larger data-driven synthetic datasets.

Furthermore, synthesising three-dimensional (3D) shapes of human anatomy has various other applications in medical restorations. Generation of high-quality artificial shapes can improve the diversity of the design of medical implant templates to benefit a more extensive range of patients in need of prosthetics and restorations. For orthopaedics implant design, Shen et al. [21] use the symmetric nature of the skeletal system to build patient-specific repaired pelvis models and manufacture the implants. However, symmetry information is not always accessible. In an orthodontics application, Barone et al. [22] utilise a combination of dental CAD templates and patient-specific jaw anatomy to synthesise 3D shapes of the individual missing tooth. More recently, Kodym et al. [23] proposed a deep learning-based 3D shape reconstruction method that generates a variety of synthetic shapes for skull defects at various stages of treatment to restore the protective and aesthetic function of the patient's skull.

1.3 Virtual Population Modeling

As discussed in Section 1.2, machine learning methods combined with computational modelling and simulation bring the possibility of gaining valuable information about new therapies and medical devices through ISCTs. Therefore, virtual populations of anatomical shapes (typically represented as computational meshes) are a key enabler for conducting ISCTs of clinical devices. More specifically, ISCTs require the generation of virtual patient populations that effectively capture significant anatomical and physiological variability, representative of target patient populations. This enables a meaningful in-silico assessment of device performance. The present study focuses on the challenge of generating representative populations of anatomical shapes.

Medical data acquisition and processing often require some manual intervention. Most techniques aim to provide a spatial description of segmented organs of interest and, typically, transform them into meshes which can be incorporated into models that are patient-specific or representative of large populations [24]. Clinical image segmentation is a time-consuming process and can be affected by observer variability. Even though machine learning approaches have shown promising results [25, 26], these models do not have high generalization ability if the object of interest is not frequent in the population. Therefore, it is concluded that building rich or descriptive generative shape models from inconsistent anatomical structures is challenging for several reasons.

First, because, real-world anatomical shapes derived from different subjects do not, in general, have topological correspondence. This poses a significant challenge for existing techniques to generate coherent, anatomically plausible shape populations. In addition, most techniques demand access to large volumes of training data containing the same semantic parts or shapes in each training sample. The process would require expensive and laborious annotation of medical imaging data to ensure anatomical shapes of interest can be accurately extracted from each subject or sample. Also, it can still be computationally challenging to create synthetic samples that are clinically meaningful and fully represents the characteristics of each individual patient. Therefore, this study aims to address the problem of generating virtual patient cohorts of anatomical geometries (e.g. Left Ventricle (LV), liver), these models allow us to perform in-silico clinical trials on the so-called digital twins [26, 27]. The approaches used to build virtual populations can be categorised into *data-driven* and *clinically-driven* methods. The descriptive statistics techniques that rely on the geometrical variability

of the samples, refer to data-driven [28]. In contrast, clinically-driven approaches require virtual cohorts with underlying clinical characteristics and typically depend on the anatomical or functional properties of the organ of interest [29]. These approaches are further discussed in Chapter 2.

1.4 Shape Representation

To construct shape models of human anatomy, organs first need to be segmented from the medical volume data and then transformed into a processible shape representation. Structured (triangulated meshes) and unstructured point-based representations of shape are the most popular for constructing shape models, due to their simplicity and flexibility. The type of shape representation also influences the choice of method for registering shapes, modelling shape variability, and eventually shape generation.

Shape as defined by [30], refers to the geometric information remaining after an object has been normalized in terms of rotation, scaling, and translation. In the literature, shapes have been represented using a variety of data structures, for instance, Point Distribution Models (PDM) [31], volumetric signed distance functions [32], point sets [33], or oriented point clouds [34]. PDM is the simplest and most generic method used for representing shapes and is comprised of a collection of points distributed over an object's surface, as well as the connectivity between the points. Early attempts to train PDMs relied on manually specified landmarks that delineated shapes of interest. However, such an approach is prohibitively expensive for large data sets with 3D anatomical structures. The point set results from a procedure that reduces the density of points on the object surface in the voxel image. A triangulation of the point set defines a surface mesh and can be used to cover the surface area in between the points. Indeed, surface meshes are composed of polygonal faces, often triangles, that cover an object's surface. Although point cloud learning has shown remarkable results in shape representation, the topology of shape is neglected, which poses significant challenges in the 3D shape understanding problem and cannot guarantee the structural accuracy of the generated results. However, this study is motivated by a more compact yet powerful way of representing shape using triangulated surface meshes that can be naturally modelled via graphs with arbitrary connectivities. Presenting shapes as graphs involves representing the geometry and connectivity of shapes using graph structures. In this representation, the vertices of the graph correspond to specific points on the

shape, and the edges capture the relationships between these points. Modelling the triangular meshes as graphs allows learning surface features such as smoothness and curvatures via edge information. In general, graphs represent shapes with structured connectivity, capturing spatial relationships, while point clouds lack this structural information, making them less suitable for shape analysis and generation. Graph-based representations are particularly powerful in capturing both local and global characteristics of shapes. They offer a flexible framework for applying graph-based algorithms and techniques for shape analysis, matching, and generation.

A triangular mesh presented as a graph is a representation of a three-dimensional surface in which each vertex of the mesh corresponds to a node in the graph, and each edge of the mesh forms an edge in the graph. The triangular mesh is composed of a collection of triangles, where each triangle is defined by three vertices connected by edges. In the corresponding graph representation, the nodes represent these vertices, and the edges of the graph connect adjacent vertices, essentially capturing the connectivity and spatial relationships of the surface. This graph-based representation allows for efficient and structured analysis of the mesh and is commonly used in various shape analysis and modelling applications.

Traditional graph generative models focus mainly on modelling the statistical properties of the graphs such as degree distributions [35–40]. Due to their simplicity, these approaches are not capable of constructing specific graphs representing certain properties such as water-tightness of meshes representing surfaces derived from organs of the human body. Hence, developing data-driven mesh generation techniques that learn directly from a set of observed meshes, modelled as graphs, remains as an open challenge.

1.5 Thesis Contributions and Overview

Given that the morphology of organs across a population is highly heterogeneous, modelling and generating these shape variations in a virtual population is a challenging task. The increasing availability of large-scale medical imaging datasets has enabled these underlying shape variations in the population to be modelled more accurately. Due to the computational challenges of working with large sample sizes, traditional methods for shape analysis can be limited in practical application [8].

In the field of shape processing, finding a meaningful shape correspondence is a

fundamental shape analysis task and is considered a challenging problem in numerous geometry processing applications, including shape generation [41]. Hence, more specifically, the aim is to develop a shape generative model that can be trained using a population of surface meshes with no correspondence and inconsistent structures (including the number of vertices and their connections) without employing any pre-processing step. This thesis addresses the data-driven generative shape modelling (virtual population modelling) with the two following specific objectives:

- Targeting an important challenge of processing shapes and presenting a dense correspondence between shapes when there are topology differences between them. A novel unsupervised deep learning framework that can directly learn correspondences from a population of spatially aligned training meshes having variable topology in anatomical structures across patients/input samples.
- Presenting a data-driven generative model for generating virtual populations of anatomical shapes that capture sufficient variability while remaining plausible. A framework with the capability to generate a virtual population of realistic synthetic shapes/meshes with variable mesh topology in anatomical structures. This approach enables the generation of anatomical shapes that are both diverse and realistic, facilitating a more comprehensive exploration of anatomical variability in virtual populations.

The proposed solutions to the objectives as mentioned above, are presented in Chapters 4 and 5, which represent the specific contributions. Each of these chapters is an adaptation of the articles that are under review, or already published in a peer-reviewed conference/journal papers.

The rest of this thesis is organized as follows:

- Chapter 2 focuses on exploring geometric deep learning methods and provides a comprehensive review of existing research on shape generation and shape matching. This chapter specifically examines the approaches used for generating shapes, with and without graph representations.
- Chapter 3 presents a baseline generative shape model based on traditional techniques. Furthermore, the chapter presents the datasets and evaluation metrics used in the study, ensuring a comprehensive and thorough evaluation of the performance of the generative shape model. By providing such a detailed overview,

this chapter sets the stage for subsequent chapters in the study, where the generative shape model is utilised to address specific research questions and objectives.

- Chapter 4 introduces a novel unsupervised deep generative model designed to generate virtual anatomical shapes by leveraging geometry and surface features from different-sized training shapes. The framework presented in this chapter employs an innovative approach by leveraging graph convolutional neural networks and attention mechanisms to derive a learnable set of correspondences across the population. Using this framework, a population-derived atlas mesh infers in a computationally closed and efficient way.
- Chapter 5 presents an end-to-end deep learning generative framework to tackle the limitation of the model presented in Chapter 4, presenting higher specificity and generalisability. This novel framework is designed to jointly learn precise refinable shape matching and generation on 3D surface mesh data while constructing a population-derived atlas. The expansion of the model into a joint clustering generative model enhances its capabilities in comprehensive shape analysis and synthesis. By incorporating multiple atlases (which leads to variable topology modelling), the model achieves a more comprehensive representation of shape anatomy, resulting in more reliable and robust outcomes. Moreover, in anatomical modelling, there are many cases where it is necessary to model the organs/objects of variable topologies, to accommodate topological changes.
- Finally, Chapter 6 concludes the thesis and discusses the outlook and future work.

CHAPTER 2

Literature Review

2.1 Introduction

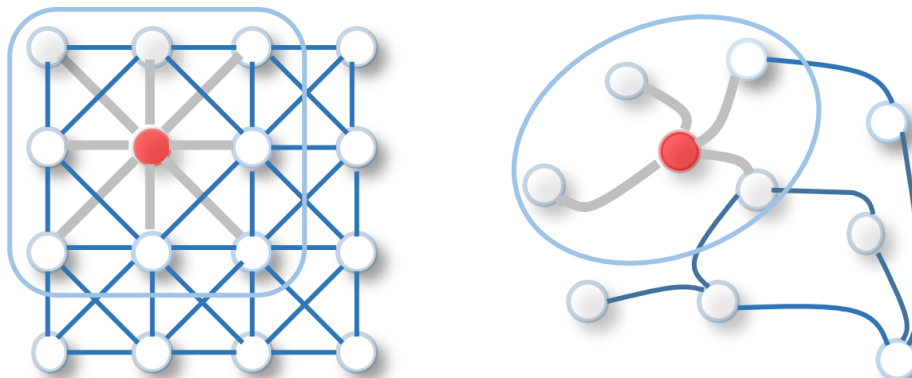
Most human anatomy exhibits considerable inter-person variability both in terms of its shape and function, which significantly impacts the effectiveness of disease prevention, diagnosis, and treatment. It is highly beneficial for clinical practice to model shape anatomy and capture this variability using data-driven methods, as this enables population-specific shape analysis and disease diagnosis. Although traditional statistical shape models have been widely used for this purpose, deep learning has become increasingly popular lately.

Many fields in medical image analysis have recently been revolutionized by the introduction of deep neural networks [42]. These approaches are able to learn complex, hierarchical feature representations that are superior to hand-crafted features in a wide range of medical imaging applications. In shape analysis, learning a shape representation may also offer advantages in shape analysis when compared to working with predefined parameterisations like point distribution models [31], medial representations [43], and diffeomorphisms [44]. The success of neural networks in image analysis is mainly due to the use of convolutional layers that benefit from the shift-invariance properties of images [45]. In spite of this, deep networks are still largely unexplored in medical shape analysis, mainly because point clouds and meshes do not have a Euclidean or grid-like structure underlying them.

This chapter discusses deep learning techniques in a non-Euclidean domain, also known as *geometric deep learning*. The technical background in geometric deep learning is covered in section.2.2, followed by a review of the existing work on general shape generation with and without graph representations.

2.2 Deep Learning on Non-Euclidean Data

The majority of the deep learning achievements over recent years have relied heavily on properties of convolutional neural networks [46]: local connectivity, weight sharing, and shift-invariance. Since those layers are defined on inputs with a grid-like structure, they cannot be easily applied to non-Euclidean domains such as discrete manifolds or (embedded) graphs. However, a large amount of data in the real world can be naturally represented as irregular structures, e.g. graph data or meshes. By applying traditional convolutional neural networks to this kind of data, a number of relevant



(a) Convolution on Euclidian data (e.g. 2D image). Pixels in an image are treated as nodes in a graph, with filter sizes determining their neighbours. The 2D convolution uses the weighted average of the red node's pixel values, along with those of its neighbours where the neighbours of a node are ordered and have a fixed size.

(b) Convolution on non-Euclidian data (e.g. graph). The graph convolutional operation can be simplified by taking the average of the node features of the red node and its neighbours to get a hidden representation of the red node. Different from Euclidian data, there is no order in the neighbours of a node, and the size of the neighbours is variable.

Figure 2.1: Illustration examples of convolution on (a) Euclidian data and (b) non-Euclidian data [2].

tasks can be improved significantly. As an emerging field, graph networks have had an enormous impact on many technological domains. A large amount of information coming from disciplines such as chemistry, biology, genetics, and healthcare is not suitable for vector-based representation and instead requires complex data structures. Due to their inherent ability to capture relationships between entities, graphs can be extremely useful in many of these applications where relational information needs to be encoded between variables. Therefore, special attention has been paid to the generalisation of Graph Neural Networks (GNN), (also called geometric deep learning), into non-structural (unordered) and structural (ordered) situations. Recently, deep learning on graphs has become an interesting research topic [2] with helpful applications in the molecule and social network analysis [47, 48], 3D mesh classification and shape correspondence [49], modelling the behaviour of dynamic interacting objects [50] and graph

generative models [51–53]. A graph convolution, for example, can be generalized from a 2D convolution. As shown in Figure 2.1, an image is a special case of a graph in which pixels are connected by adjacent pixels. Graph convolution can also be performed by averaging neighbourhood information for nodes, similar to 2D convolution.

2.2.1 Graph Construction

Graphs are non-Euclidean data structures that model a set of vertices and their relationships via edges. Many complex data structures in the real world can be naturally represented as graphs. Hence, the data-driven generation of synthetic graphs will have an exciting impact in various fields including computer vision and computational anatomy.

Mathematically, a graph is represented as $G = (V, E)$ where V is a set of nodes and E is the set of edges, and $N = |V|$, $M = |E|$. Let $v_i \in V$ denote a node and $e_{ij} = (v_i, v_j) \in E$ denote an edge. The adjacency matrix \mathbf{A} is derived as a $N \times N$ matrix with $A_{ij} = 1$ if $e_{ij} \in E$ and $A_{ij} = 0$ if $e_{ij} \notin E$. A graph may have node attributes \mathbf{X} , where $\mathbf{X} \in \mathbb{R}^{N \times D}$ is a node feature matrix with $\mathbf{x}_v \in \mathbb{R}^D$ representing the feature vector of a node v . Meanwhile a graph may have edge attributes \mathbf{X}^e , where $\mathbf{X}^e \in \mathbb{R}^{M \times C}$ is an edge feature matrix with $\mathbf{X}_{v,u}^e \in \mathbb{R}^C$ representing the feature vector of an edge (v, u) .

The Laplacian matrix of a graph is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where \mathbf{D} is a diagonal matrix of node degrees and $D_{ii} = \sum_j A_{i,j}$. It is common to present the Laplacian matrix in three following forms, Combinatorial Laplacian:

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \tag{2.1}$$

Symmetric Normalized Laplacian:

$$\mathbf{L}^{sym} = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}} = \mathbf{I}_N - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \tag{2.2}$$

and Random Walk Normalized Laplacian:

$$\mathbf{L}^{rw} = \mathbf{D}^{-1} \mathbf{L} = \mathbf{I}_N - \mathbf{D}^{-1} \mathbf{A} \tag{2.3}$$

The Laplacian matrix \mathbf{L} is symmetric and positive-semidefinite for an undirected graph G , and dimension of the eigenspace is N . For each of the three forms of Laplacian

matrix, the representation of the elements of matrix is given by:

$$\mathbf{L}_{i,j} = \begin{cases} \text{deg}(v_i) & \text{if } i = j, \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j, \\ 0 & \text{otherwise.} \end{cases} \quad (2.4)$$

where $\text{deg}(v_i)$ is the degree of the vertex i . The elements of \mathbf{L}^{sym} are defined by

$$\mathbf{L}_{i,j}^{sym} = \begin{cases} 1 & \text{if } i = j \text{ and } \text{deg}(v_i) \neq 0, \\ \frac{-1}{\sqrt{\text{deg}(v_i)\text{deg}(v_j)}} & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j, \\ 0 & \text{otherwise.} \end{cases} \quad (2.5)$$

Next, the elements of \mathbf{L}^{rw} are given by:

$$\mathbf{L}_{i,j}^{rw} = \begin{cases} 1 & \text{if } i = j \text{ and } \text{deg}(v_i) \neq 0, \\ \frac{-1}{\text{deg}(v_i)} & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j, \\ 0 & \text{otherwise.} \end{cases} \quad (2.6)$$

The normalized graph Laplacian matrix possesses the property of being real symmetric positive semidefinite. With this property, the normalized Laplacian matrix can be decomposed as $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where $\mathbf{U} \in \mathbb{R}^{N \times N}$ is the matrix of eigenvectors ordered by eigenvalues, $\mathbf{\Lambda}$ presents the diagonal matrix of eigenvalues (spectrum), where $\Lambda_{ii} = \lambda_i$, $0 = \lambda_0 \leq \lambda_1 \leq \dots \leq \lambda_{N-1}$. The eigenvectors of the normalized Laplacian matrix form an orthonormal space, in mathematical words, $\mathbf{U}^T\mathbf{U} = \mathbf{I}$.

Since the Laplacian matrix is a discrete Laplacian operator, it is natural to use the Laplacian matrix and its eigenvectors to define the Fourier transform of the graph in the spectral method. In graph signal processing a graph signal $\mathbf{x} \in \mathbb{R}^N$ is a feature vector of all nodes of a graph. The graph Fourier transform to a signal \mathbf{x} is defined as $\mathfrak{F}(\mathbf{x}) = \mathbf{U}^T\mathbf{x}$ also, the inverse graph Fourier transform is defined as $\mathfrak{F}^{-1}(\hat{\mathbf{x}}) = \mathbf{U}\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}$ denotes the resulted signal from the graph Fourier transform. The graph Fourier transform projects the input graph signal to the orthonormal space where the basis is formed by eigenvectors of the normalized graph Laplacian.

2.2.2 Graph Convolutional Neural Networks

In recent years, geometric deep learning [45] has emerged as a set of methods that aims to extend existing neural networks through graph theory and define convolution operations for deep neural networks that can handle irregular inputs. In order to

generalize the traditional convolution operation from images or grids to graphs, the key is to learn a function f which is capable of aggregating the features \mathbf{x}_i from node v_i and its neighbours v_j 's features \mathbf{x}_j . Thus, a new representation of node v_i can be presented. For the l -th graph convolution layer, we denote the input node representations of all nodes by the matrix $\mathbf{Y}^{(l-1)}$ and the output node representations $\mathbf{Y}^{(l)}$.

The existing work in this field can be broadly categorized into two subsets: *Spectral* and *Spatial* filtering approaches. The former is based on spectral graph theory [54], in which the eigenvalues of a graph's Laplacian matrix are interpreted as node frequencies [55]. Similar to Fourier domain filtering of traditional signals, they are filtered in the spectral domain, where the graph convolutional operation is interpreted as removing noise from graph signals. In the latter subset, the spatial approaches, convolution is performed in local Euclidean neighbourhoods w.r.t local positional relations between points. Where for example polar, spherical and Cartesian coordinates can be used to represent local positional relations between nodes. In this subsection, we will focus on the technical background of major spectral-based and spatial-based graph neural networks.

2.2.2.1 Spectral-based Graph Convolutional Networks

The convolution theorem states that the Fourier transform of convolution is the product of Fourier transforms, by analogy to the graph and puts in the definition of the Fourier transform on the graph. Now, the graph convolution of the input signal \mathbf{x} with a filter $\mathbf{g} \in \mathbb{R}^N$ is defined as:

$$\mathbf{x} *_G \mathbf{g} = \mathfrak{F}^{-1}(\mathfrak{F}(\mathbf{x}) \odot \mathfrak{F}(\mathbf{g})) = \mathbf{U} \left(\mathbf{U}^T \mathbf{x} \odot \mathbf{U}^T \mathbf{g} \right) \quad (2.7)$$

where \odot denotes the Hadamard product (element-wise). If we denote a filter as $\mathbf{g}_\theta = \text{diag}(\mathbf{U}^T \mathbf{g})$, then the spectral graph convolution is simplified as

$$\mathbf{x} *_G \mathbf{g}_\theta = \mathbf{U} \mathbf{g}_\theta \mathbf{U}^T \mathbf{x} \quad (2.8)$$

We can understand \mathbf{g}_θ as a function of the eigenvalues of \mathbf{x} , i.e. $\mathbf{g}_\theta(\boldsymbol{\Lambda})$. Evaluating Equation 2.8 is computationally expensive, due to multiplication with the eigenvector matrix \mathbf{U} ($O(|V|^2)$). The Spectral-based graph convolutional networks all follow the definition in Equation 2.8, and the most critical difference between the different frameworks lies in the design of filter \mathbf{g}_θ .

Bruna *et al.* [56] proposed the first spectral-based graph convolutional neural network (Spectral GCN), which assumed the filter $\mathbf{g}_\theta = \Theta_{i,j}$ is a diagonal matrix filled with learnable parameters. The graph convolutional layer of Spectral GCN is defined as

$$\mathbf{Y}_{:,j}^{(l)} = \sigma \left(\sum_{i=1}^{f_{l-1}} \mathbf{U} \Theta_{i,j}^{(l-1)} \mathbf{U}^T \mathbf{Y}_{:,i}^{(l-1)} \right); (j = 1, 2, \dots, f_l) \quad (2.9)$$

where l is the layer index, f_{l-1} is number of input channels and f_l is number of output channels. $\mathbf{Y}^{(l-1)} \in \mathbb{R}^{N \times f_{l-1}}$ represents the input graph signal, $\mathbf{Y}^{(0)} = \mathbf{X}$, and σ is a non-linearity function (e.g. Rectified Linear Unit (ReLU)) applied on the vertex-wise function values. Though Spectral GCNs make significant theoretical contributions, they have several major drawbacks due to the eigen decomposition of the Laplacian matrix. First, the eigenbasis of a graph changes with a perturbation. Second, due to the coefficients of each spectral filter being based on the basis and the Laplacian of a graph, the learned filters can not be applied to graphs with different structures. Third, since a filter defined in the spectral domain is not naturally localized, translations are costly (i.e. $O(|V|^3)$) due to the $O(|V|^2)$ multiplication with the graph Fourier basis. In further works, ChebNet [57] and GCN* [58]¹, several approximations and simplifications are made to reduce the computation complexity to $O(|E|)$.

In general, any polynomial of order k can be expanded with a Chebyshev polynomial. Defferrard *et al.* [57] proposed ChebNet framework that approximates the filter \mathbf{g}_θ by Chebyshev polynomials of the diagonal matrix of eigenvalues ($\mathbf{g}_\theta = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{\mathbf{\Lambda}})$), where $\tilde{\mathbf{\Lambda}} = 2 \frac{\mathbf{\Lambda}}{\lambda_{max}} - \mathbf{I}_N \in [-1, 1]$ and λ_{max} denotes the largest eigenvalue of \mathbf{L} . $\boldsymbol{\theta} \in \mathbb{R}^K$ is now a vector of Chebyshev coefficients. The Chebyshev polynomials in recursive form are defined by $T_k(\mathbf{x}) = 2\mathbf{x}T_{k-1}(\mathbf{x}) - T_{k-2}(\mathbf{x})$ with $T_0(\mathbf{x}) = 1$ and $T_1(\mathbf{x}) = \mathbf{x}$. Thus the convolution of filter \mathbf{g}_θ and signal \mathbf{x} is defined as:

$$\mathbf{x} *_G \mathbf{g}_\theta = \mathbf{U} \left(\sum_{k=0}^{K-1} \theta_k T_k(\tilde{\mathbf{\Lambda}}) \right) \mathbf{U}^T \mathbf{x} = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{\mathbf{L}}) \mathbf{x} \quad (2.10)$$

where $\tilde{\mathbf{L}} = 2 \frac{\mathbf{L}}{\lambda_{max}} - \mathbf{I}_N$. This expression is now K -localized, as it is a K^{th} -order polynomial in the Laplacian, which means that it is dependent only on nodes that are at maximum K steps away from the central node (K -hop neighbourhood). Therefore, the complexity of evaluating Equation 2.10 is $O(|E|)$, i.e. linear in the number of edges.

¹As GCN is used to represent broad graph convolutional neural networks in the study, we name this particular method GCN* to avoid ambiguity.

In Graph Convolutional Network (GCN*) [58], Kipf *et.al.* introduced a first-order approximation of ChebNet (i.e. operating filters on 1-hop neighbourhoods of the graph). Assuming $K = 1$ and $\lambda_{max} = 2$, Equation 2.10 is simplified as

$$\mathbf{x} *_G \mathbf{g}_\theta = \theta_0 \mathbf{x} - \theta_1 \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{x} \quad (2.11)$$

GCN further considers a unified hypothesis $\theta = \theta_0 = -\theta_1$, to prevent overfitting due to an excessive number of parameters; leading to the following definition of graph convolution

$$\mathbf{x} *_G \mathbf{g}_\theta = \theta (\mathbf{I}_n + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}) \mathbf{x} \quad (2.12)$$

To use multi-channels of inputs and outputs in GCN operation, Equation 2.9 can be modified as

$$\mathbf{Y} = \mathbf{X} *_G \mathbf{g}_\theta = f(\bar{\mathbf{A}} \mathbf{X} \Theta) \quad (2.13)$$

where $\bar{\mathbf{A}} = \mathbf{I}_N + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ and $f(\cdot)$ is an activation function. Using $\mathbf{I}_n + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ empirically causes numerical instability to GCN. To address this problem, GCN applies a normalization trick to replace $\bar{\mathbf{A}} = \mathbf{I}_N + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ by $\bar{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}}$ with $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ and $\tilde{D}_{ii} = \sum_j \tilde{A}_{i,j}$. Therefore the layer-wise propagation rule in Equation 2.14 can be written as:

$$\mathbf{Y}^{(l)} = f\left(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{Y}^{(l-1)} \Theta^{(l-1)}\right) \quad (2.14)$$

where $\Theta^{(l-1)} \in \mathbb{R}^{f_l \times f_{l-1}}$ is a layer-specific trainable weight matrix, $\mathbf{Y}^{(l-1)} \in \mathbb{R}^{N \times f_{l-1}}$ and $\mathbf{Y}^{(l)} \in \mathbb{R}^{N \times f_l}$ represent the input and output graph signals respectively.

Further, Li *et.al.* [59] showed that GCN* convolutions are simply a form of Laplacian smoothing, which combines the features of a vertex and its neighbours. The smoothing operation makes the features of vertices in the same cluster similar, which is the key reason why GCN*s present promising results, but it also brings potential concerns of oversmoothing with many convolutional layers.

2.2.2.2 Spatial-based Graph Convolutional Networks

A spatial convolution refers to the direct execution of convolution operations on a graph. Similar to the convolutional operation of a conventional CNN on an image, spatial-based methods define graph convolutions based on the spatial relationships among nodes. Images can be considered as a special form of a graph with each pixel representing a

node. The pixels are directly connected to their neighbours, as shown in Figure 2.1(a). A filter is applied to a 3×3 patch by taking the weighted average of pixel values of the central node and its neighbours across each channel. Similarly, in spatial-based graph convolutions, the representation of the central node is convolved with the representation of its neighbours to obtain an updated representation of the central node as shown in Figure 2.1(b).

Considering that the size of traditional convolution kernels is fixed, a neighbourhood of a fixed size is required to perform traditional convolution on a graph. However, as graphs are irregular structured data, nodes in a graph usually have different numbers of neighbourhoods as opposed to data with a regular grid structure. From a spatial perspective, GCN can be viewed as collecting feature information from the neighbourhood of a node. Therefore, Equation 2.13 in node states can be expressed as

$$\mathbf{y}_v = f \left(\left(\sum_{u \in \{\mathcal{N}(v) \cup v\}} \bar{\mathbf{A}}_{v,u} \mathbf{x}_u \right) \Theta \right) \quad \forall v \in V. \quad (2.15)$$

where $\mathcal{N}(v)$ illustrates the neighbors of a node v .

In spatial-based approaches, graph convolutions are defined by aggregating information. The idea has been inherited from recurrent-based Graph Neural Networks (e.g. [60]), where each node exchanges information with its neighbours until equilibrium has been reached. The first work towards spatial-based GCNs is the Neural Network For Graphs (NN4G) [61]. NN4G performs graph convolutions by summing up the neighbourhood information of each node directly. Over each layer, residual connections and skip connections are also used to memorize information. As a result, the spatial-based convolution of NN4G is defined as

$$\mathbf{Y}^{(l)} = f \left(\mathbf{XW}^{(l-1)} + \sum_{i=1}^{l-1} \mathbf{AY}^{(i-1)} \Theta^{(l-1)} \right), \quad (2.16)$$

which resembles the form of GCN* [58]. In contrast to GCN*, NN4G uses an unnormalized adjacency matrix, which may potentially lead to numerical instability.

Mixture Model Network (MoNet) [62] is a spatial-domain method to extend CNN architectures on non-Euclidean domains such as graphs and manifolds. This approach formulates convolutions as template matchings with local intrinsic *patches* on graphs or manifolds. Some spatial-based methods disregard the relative positions between nodes and their neighbours when integrating neighbor's information. To address this, MoNet

assigns different weights to a node's neighbours and introduces a pseudo-coordinate system to determine the relative position between a node and its neighbour. Once the relative position between two nodes is known, a weight function maps the relative position to the relative weight between these two nodes. In this framework, for each vertex v , a d -dimensional vector of pseudo-coordinates $\mathbf{u}(x_v, x_{v'})$ correlates v and points $v' \in \mathcal{N}(v)$ in its neighborhood. It constructs the *patch* operator as the following general form:

$$D_j(x_v)f = \sum_{v' \in \mathcal{N}(v)} w_j(\mathbf{u}(x_v, x_{v'})) f(v'), \quad j = 1, \dots, J \quad (2.17)$$

where J represents the dimensionality of the extracted patch, $\mathbf{W}_\Theta(\mathbf{u}) = (w_1(\mathbf{u}), \dots, w_J(\mathbf{u}))$ is a weighting function (kernel) parameterized by learnable parameters Θ , and each Gaussian kernel is defined as

$$w_j(\mathbf{u}) = \exp\left(\frac{-1}{2}(\mathbf{u} - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{u} - \boldsymbol{\mu}_j)\right), \quad (2.18)$$

where $\boldsymbol{\Sigma}_j$ and $\boldsymbol{\mu}_j$ are learnable $d \times d$ and $d \times 1$ covariance matrix and mean vector of a Gaussian kernel, respectively. Equations 2.17 and 2.18 can thus be interpreted as a Gaussian Mixture Model (GMM). Thus, the generalised form of the spatial convolution on non-Euclidean domains is given by using a template-matching process:

$$(f * g)(x_v) = \sum_{j=1}^J g_j D_j(x_v)f. \quad (2.19)$$

$D_j(x_v)f$ can be regarded as a *patch* on the graph and template g is the filter acting on patch. Under the MoNet framework, several existing approaches for manifolds such as Geodesic Convolutional Neural Network (GCNN) [63], SplineCNN [49], and for graphs such as GCN* [58], Diffusion-Convolutional Neural Network (DCNN) [64] can be generalized as special instances of MoNet by constructing non-parametric weight functions. MoNet additionally proposes a Gaussian kernel with learnable parameters to learn the weight function adaptively. SplineCNN [49] follows the same framework (i.e. Equation 2.19), but uses a different convolution kernel based on B-splines.

Local filters [62, 63] are defined by hard-coded local pseudo-coordinates over the graph, which may not be optimal. As opposed to previous works, Verma *et.al* [3], further introduced Feature-Steered Network (FeaStNet), a spatial-based graph convolution operator that exploits the learning features from the preceding network layer to dynamically determine the association between filter weights and graph neighbourhood, rather than relying on static predefined local pseudo-coordinate systems.

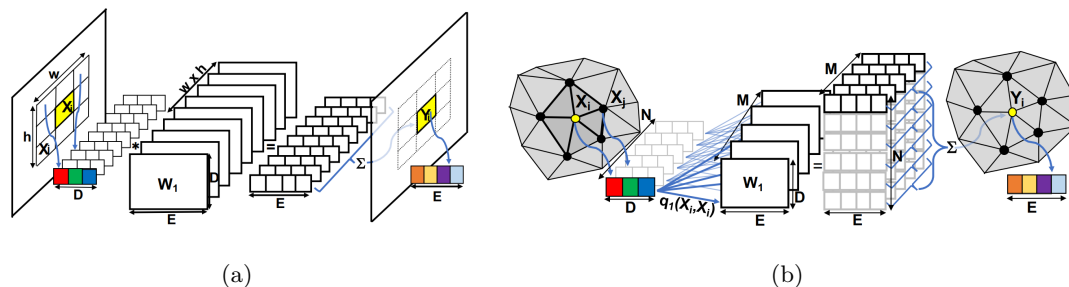


Figure 2.2: (a) Illustration of a standard CNN, representing the parameters as a set $M = w \times h$ weight matrices. Each weight matrix is associated with a single relative position in the input patch. (b) Feature-Steered graph convolution network [3], where each node (e.g. i) in the input patch is softly associated to each of the M weight matrices based on their features using $q_m(x_i, x_j)$.

In traditional Convolutional Neural Networks (CNNs), the convolutional filter weights are defined in a set of $M = w \times h$ weight matrices $\mathbf{W}_m \in \mathbb{R}^{E \times D}$, where each weight matrix is utilized to project input features $\mathbf{x} \in \mathbb{R}^D$ to output features $\mathbf{y} \in \mathbb{R}^E$ (see Figure 3.2(a)). Thus pixel-wise convolution can be expressed as follows:

$$\mathbf{y}_v = \mathbf{b} + \sum_{m=1}^M \mathbf{W}_m \mathbf{x}_{n(m,v)}, \quad (2.20)$$

where $\mathbf{b} \in \mathbb{R}^E$ denotes a vector of bias terms, and $n(m, v)$ gives the index of the neighbor in the m -th relative position w.r.t pixel v . In FeaStNet [3], a soft correspondence map (/assignment) $q_m(x_v, x_j)$ is made for the j -th neighbour of a node v , rather than assigning each neighbour to a single weight matrix (see Figure 3.2(b)). Therefore, the output of the convolutional operator can be defined as

$$\mathbf{y}_v = \mathbf{b} + \sum_{m=1}^M \frac{1}{|\mathcal{N}_v|} \sum_{j \in \mathcal{N}_v} q_m(x_v, x_j) \mathbf{W}_m \mathbf{x}_j, \quad (2.21)$$

where a translation-invariant correspondence operator $q_m(x_v, x_j)$ learns to dynamically assign x_j to the m -th learnable weight matrix \mathbf{W}_m , using a soft-max over a linear transformation of the local feature vectors,

$$q_m(x_v, x_j) \propto \exp(\mathbf{u}_m^T \mathbf{x}_v + \mathbf{p}_m^T \mathbf{x}_j + \mathbf{c}_m), \quad (2.22)$$

where \mathbf{u}_m , \mathbf{p}_m and \mathbf{c}_m are parameters of the linear transformation. Further in Chapter

4, this spatial-based graph convolution operator is utilized to establish correspondences between shapes.

2.2.2.3 Comparison between the Spectral and Spatial convolutional filtering approaches

In this section, the differences between the spectral and spatial convolutional approaches are discussed as follows:

First, the spectral approaches all assume information is encoded only in connectivity, edge weights, and node features. Although this may be true for general graphs, it does not hold for embedded graphs or meshes, where the relative positions of nodes provide additional information, which can be taken into account in Spatial-based methods. Second, spectral filtering approaches rely on the eigen-decomposition of the graph Laplacian. However, this decomposition is often unstable, any perturbations to a graph lead to a change of eigenbasis, making generalizations across different shapes challenging [62]. In contrast, spatial-based models perform graph convolution locally on each node, making it possible to share weights across multiple locations. Third, spectral-based models are less efficient than spatial-based models. Due to the computation cost of spectral-based models changing with graph size, they are not scalable to large graphs or parallelizable. Moreover, for the spectrorization-based method, the eigen-decomposition operation is required, which has a significant impact on the time and computational costs. In contrast, spatial-based approaches enable models to deal with large graphs and perform convolution operations directly on graphs by clustering neighbouring nodes. Finally, the most important difference exploited in this study is the ability of spatial-based methods to handle inconsistent graph populations, while most spectral-based approaches are limited to fixed graph structures.

2.3 Generative Shape Modeling

Learning graph generative models is a challenging task for deep learning and has wide applicability to a range of domains. However current deep neural methods suffer from limited scalability. Developing an effective graph generative model depends on representing a distribution over graphs. Basically, the idea behind the representation learning approaches is to learn a mapping that embeds nodes, or entire (sub)graphs, as points

in a low-dimensional vector space \mathbb{R}^d . The goal is to optimize this mapping so that geometric relationships in the embedding space reflect the structure of the original graph. After optimizing the embedding space, the learned embeddings can be used as feature inputs for downstream machine learning tasks. In addition, real-world graphs are not inherently corresponding which makes it challenging to process due to different topological structures.

Generative shape modelling using surface representations often involves two fundamental aspects of *Shape matching (or Establishing correspondences)* and *Generation*. More specifically, the framework follows establishing point-to-point spatial correspondences between various shapes (hence deriving a consistent representation across the training data) and deriving a generator that learns from the unified shape representations and samples new instances. Hence, to review the related works, first, these two aspects are investigated.

2.3.1 Establishing Correspondences

Shape matching typically refers to the problem of establishing a set of meaningful point-wise correspondences between shapes, which is often a difficult task because the structures need to be understood at both the local and global scales.

A large body of research has been dedicated to registration of shapes as point sets. However, the majority of these methods restrict generalization to input point sets with identical cardinality. These works can be broadly categorized into three different groups:

The first group of methods [65, 66] model the point sets as instantiations from underlying probability distributions and minimize a distance over those distributions to derive the spatial transformations between various shapes. These methods use soft-assignment of correspondences, which means assigning correspondences between all points according to some probability. Coherent point drift (CPD) [65] is a well-known algorithm for point set registration, based on Gaussian mixture model (GMM). To evaluate the transformation, CPD imposes a motion coherence constraint over the velocity field. This algorithm provides the possibility of performing matching on a subset of the points while computing the transformation. CPD model's performance is restricted by a number of commonly known obstacles. Firstly, deformation involving a high degree of freedom is difficult to model efficiently. Moreover, due to the complexity

of deformations, the optimization can easily get stuck in local optima. The Procrustes method is also used for shape analysis in [67]. Given the correspondences, this method will align two sets of landmarks. In contrast, our method does not require initial landmarks to establish correspondences between pairs of shapes.

The second class of methods aim to estimate point correspondences during the alignment process. These different family of registration methods, in contrast to the first class (e.g. CPD), use binary assignments of correspondences. Besl *et al.* [68] proposed a simple and efficient method for 3D shape registration. The Iterative Closest Point (ICP) algorithm is used to estimate correspondences based on the closest geometric neighbour, while the algorithm converges to a local optimal alignment between two point clouds. Although the standard ICP algorithm is an efficient method in a wide range of applications, it suffers from the slow convergence and sensitivity to initial alignment, noise, outliers and partial overlap. Research in [69] has been presented to address these problems. The "Deep ICP" method in [70] represented an advanced extension of the traditional Iterative Closest Point algorithm, leveraging deep learning techniques. In deep ICP, neural networks are employed to enhance the accuracy and robustness of point cloud registration. The method combined the iterative refinement principle of ICP with the representational power of deep neural networks, allowing for more effective handling of complex and non-rigid deformations in 3D point clouds. In contrast, we consider the structures of the shapes within the context of the graphs that can benefit from connectivities over both local and global scales to derive reliable correspondences between a pair of shapes. Unlike a local-search algorithm such as ICP, our method utilized both local and global contexts based on features that are extracted using a Graph Convolutional Network (GCN) and an attention module.

The third class involves the methods that represent correspondences simply as relationships between two datasets, without considering the geometric features. These approaches estimate a pairwise assignment between vertices using graph matching. The proposed method in [71] models correspondences in a probabilistic manner and computes a vertex-to-vertex probabilistic assignment. Another group of methods, proposed in [72, 73] make use of the deterministic optimization technique that explores the quadratic assignment problem to establish correspondence between two point sets.

More recent works on graph matching are based on deep learning methods to find the optimal point-to-point correspondences [74–76]. These approaches develop supervised

graph matching networks based on displacement rather than a registration task. However, the proposed graph matching methods are performed offline and remain unaltered during shape generation. Further, Bai *et.al.* in [77, 78] proposed a pair-wised graph matching framework which was a learning-based framework rather than a pairwise graph distance computation. These methods, however, are only capable of calculating similarity scores between whole graphs due to their use of spectral-based graph convolution techniques. To aggregate neighbour information, they employed multi-scale GCN layers and then calculated multiple similarity matrices which increases time complexity for large-scale graphs.

2.3.2 Shape Generation

This study focuses on generating virtual populations of anatomical shape represented as a 3D surface mesh (i.e. graph). While several previous studies have proposed image-based generative models for capturing anatomical variability across populations, we restrict our discussion of relevant literature on previous studies that utilize statistical or generative shape modelling

Statistical Shape Models (SSMs) are established tools to capture morphological variations over a training set using point distribution models. A straightforward method to statistically gather considerable natural variability of training shapes, making it very common to use for generating virtual populations. Most SSM-based generative approaches aimed to preserve the topology of segmented anatomical structures through model-constrained image segmentation techniques, using SSMs as shape priors. However, a detailed discussion of these methods is beyond the scope of this study; further information can be found in the review by Heimann *et al.* [79].

Synthetic anatomical shapes can be generated by sampling SSMs implementing variants of Principal Component Analysis (PCA) on shape spaces. In SSM training using (PCA), point-based representations of shape are the most prevalent used, due to their simplicity and independence from topology. Cootes *et al.* presented the best well-known SSM methods, the Active Shape models [31] and Active Appearance models [80]. PCA-based statistical shape modelling has been extensively used for generating virtual populations of anatomy [33, 81], quantitative shape analysis for computer-aided diagnosis [82] and in model-based segmentation [83, 84]. Statistical shape modelling, i.e., data-driven methods, has been used in some studies to investigate correlations

between anatomical phenotypes and shape. For instance, Cosentino *et al.* in [85] explored the aortic morphology and the associations between the shape and function of aortic valves, using shape modes linked to specific aneurysmal aortic morphological features. A SSM-based model is proposed in [86] to describe the morphological variability of the paediatric left ventricle and examine its relationship with biometric parameters, and identify obesity-associated adverse anatomical remodelling. Using a subset of landmarks as predictors, Syrkina *et al.* [87] and Blanc *et al.* [88] employed a conditional SSM to estimate the shape of the unseen landmark points. Nevertheless, these studies rely on parametric methods in which: i) shapes are parameterized by landmarks, and ii) point-to-point correspondence must be achieved between input shapes. Gooya *et al.* [33] proposed a generative model for shapes from point clouds without point correspondences using a mixture of PCA models, however, the connectivities of the points were discarded. We can summarise, a pre-requisite for PCA-based SSMs is point-wise correspondence across the population of training shapes and the same cardinality of shapes across all samples in the training population.

A number of recent advances in deep learning have demonstrated that deep neural networks can be used as powerful generative models for both images and geometric data (e.g. point clouds, meshes, graphs), as they are capable of learning rich hierarchical representations of the data [89]. A systematic approach for generative modelling is to sample synthetic instances from a probability density function (pdf) that has been fitted to observed real data. Hence, a significant body of research in this area focuses on learning a distribution of given graphs and generating synthetic graphs by Generative Adversarial Networks (GANs) [90] and Variational AutoEncoders (VAEs) [91]. These methods present a promising framework for generating data with fixed topology.

A few studies [92–95] have adopted these approaches for generating virtual populations of anatomy. Danu *et al.* [94] employed deep generative models (VAE and GAN) for generating voxelised vessel surfaces, where they represented the unstructured surface mesh as a three-dimensional image. Therefore, it becomes inherently compatible with the standard convolutional neural network architecture. Although, the results show potential on employing deep neural network based generative models on three-dimensional surfaces, they are not able to deal with the complex data structures.

More recently, a deep learning framework for anatomical shapes analysis is presented in [95], and demonstrated its application for discriminative and generative tasks. This

method proposes a conditional generative model, where the condition vector controls generative processes. This approach operates on unordered point clouds, with fixed cardinality, without requiring the correspondences between them. While the synthetic shapes appear realistic, it is difficult to determine how real the randomly generated shapes are, since only the generalization ability of the model is presented and any other criteria have not been considered to assess the virtual population. Presenting cardiac biventricular anatomy as point clouds, Beetz *et al.* [92] proposed a geometric deep learning method for generating populations of realistic biventricular anatomies. The generation of personalized anatomies is further enhanced by adding subpopulation-specific characteristics as conditional inputs. It is worth noting that these approaches rely on point-cloud representations which cannot leverage surface information encoded in meshes. Using binary masks of aortas as inputs, Romero *et al.* [93] explored the generation efficiency of the generative adversarial network (GAN). This method specifically addresses the generation of a cohort of patients meeting a specific clinical criterion without having access to a reference sample of that phenotype.

Recently, multiple methods have been proposed to generate non medical 3D shapes. Extending SSMs, Nash *et al.* propose ShapeVAEs [34] by training a VAE that learns part-segmented 3D objects from oriented point clouds. However, like conventional SSMs, the method needs a dense set of known point-to-point correspondences. Litany *et al.* in [96] present a generative model based on convolutional operators used for deformable shape completion from partial shapes. Ranjan *et al.* introduce CoMA, a model constructed from spectral graph convolutions using Chebyshev filters [97]. Due to spectral-based convolution, which relies on fixed graph structures, this method cannot be generalized to inconsistent graph populations. To generate high-quality 3D shapes, Yang *et al.* introduce DSG-Net, a VAE network that learns a disentangled structured and geometric mesh representation in a synergistic manner [98]. Deep learning based on variational autoencoders is also employed in [99] for mesh generation. All of these methods can only be applied to data with the same graph structure. Ben-Hamu *et al.* [100] proposed a GAN based 3D shape generative model by first deriving a tensor representation for genus-zero shapes using multi charts covering the shapes. However, the method requires a weak supervision in the form of prior sparse correspondences provided by the user. Moreover, unfolding the 3D shapes on image-like structures (such as the geometry images proposed in [101]) can be a complex and non-differentiable

procedure. Nash *et al.* in [102] proposed PolyGen, a generative model that is trained on 3D polygon meshes. The method converts the shapes into a sequence of vertices and uses transformer networks [103] to predict tokens in the sequence. The method shows promising results using samples available in ShapeNet datasets. However, there is no guarantee the generated meshes are closed and water-tight, which is often necessary for human organs in medical imaging applications.

Diffusion generative modelling is an innovative approach to generative modelling that focuses on optimising the diffusion process to generate data samples [104]. This technique is particularly effective for complex data distributions like images, audio, or 3D meshes. In generative models like Generative Adversarial Networks (GANs) or Variational Autoencoders (VAEs), generating samples involves iteratively refining random noise to resemble the desired data distribution. Diffusion generative modelling takes a different route by directly modelling the process of transforming a simple distribution into the desired data distribution through a diffusion process [104]. Some recent studies in [105, 106] used these techniques for generating 3D mesh models. Luo *et al.* in [106] formulate the point cloud generation as a denoising process, which is implemented as a Markov chain. For any 3D point, it trains a network to predict the point's gradient that is used to move the point to the high-density area. A method called MeshDiffusion is introduced for generative 3D mesh modelling in [105]. The method utilised a score-based generative model, which means it learns the mesh generation process directly through the gradient of a score function. This approach enables MeshDiffusion to generate high-quality 3D meshes with complex structures and detailed features. Although diffusion modelling provides a coherent framework for modelling complex data distributions, allowing for flexible and controlled generation, challenges arise in terms of training stability, memory and computational demands, as well as the interpretability of noise introduced during the diffusion process. Careful tuning of hyperparameters and addressing mode capture limitations are essential for harnessing the full potential of diffusion models in 3D shape generation tasks.

2.4 Conclusion

Graphs are used to represent shapes in this study, so geometric deep learning, which is a generalization of deep neural networks for non-Euclidean domains (such as graphs), is the first topic discussed in this chapter. A comprehensive review of two different

categories of geometric deep learning: Spectral-based and Spatial-based methods was presented. The spatial-based methods are capable of handling inconsistent graph populations, while the spectral-based approaches are limited to unified graph structures. Spatial-based models have also been shown to provide additional information (e.g. relative positions of vertices) rather than just spectral information. The important differences between spectral and spatial graph convolution methods will be exploited in this study.

This chapter also discussed generative shape modeling, which involves establishing correspondences and generating shapes. In the past few decades, a considerable amount of research has been dedicated to the registration of shapes as point sets and establishing point-to-point correspondences. These methods, however, have a number of limitations, including their ability to work only with input sets with the same cardinality, being supervised (required initial landmarks), and the use of a local search algorithm. Another group of works used graph matching to establish correspondences, most of which used displacement and were done offline and remained unchanged throughout the generation of shapes. Furthermore, a wide range of research on the generation of both medical and non-medical shapes was reviewed.

In summary, existing works on shape generative modeling using graphs are valuable but often require specific conditions, such as training datasets with identical graph connectivity and a fixed number of vertices, supervised vertex-to-vertex correspondences, and unfolding the meshes into image-like structures or a linearising of the graphs into sequences, which can be complex.

CHAPTER 3

Baseline Generative Shape Model: Datasets,
Evaluation Metrics, and Performance Analysis

The previous chapter provided a detailed review of generative shape modelling and the related challenges in this field. Based on this, this chapter proceeds to introduce the baseline generative shape model, which serves as a baseline for this study. Additionally, the datasets and evaluation metrics utilised are also presented to enable a thorough and comprehensive assessment of the performance of the generative shape model.

3.1 Datasets

The anatomical shapes are derived from real clinical images. The proposed methods are applied to two clinical datasets comprising actual 3D surface mesh datasets (represented as grid graphs) obtained from Cardiac Magnetic Resonance (CMR) and liver Computed Tomography (CT) images.

In this study, the impact of two factors on generative models is assessed: (i) the size of training data, and (ii) the variability of shapes among individuals in the population. Therefore, two datasets with a different number of shapes and various complexities in the structure are used to assess the model’s versatility and performance. A large dataset of left ventricle shapes is used because of their unique structural geometry (inner and outer wall of the left ventricle in a 3D shape). In addition, a small set of liver shape datasets is utilised where liver shapes include large morphological variations in their geometry.

Left Ventricle (LV)

LV shapes were derived from cardiac magnetic resonance (CMR) images available in the UK Biobank [107] under access application number 11350. This dataset was available in the CISTIB lab from the study in [108]. The experiments are performed on 1000 LV surface meshes prepared using the method described in [108]. All utilized LV graphs are related to the end-diastolic phase of the cardiac cycle. Mesh cardinalities vary from 1000 to 2000 points.

Liver

3D liver shapes are obtained from the public CT-ORG image dataset from The Cancer Imaging Archive (TCIA) [109]. This dataset consists of 140 CT scans, each with multi

organs labelled. These organs were automatically segmented by morphological image processing or manually by clinical experts. The images come from a wide variety of sources, including abdominal and full-body; contrast and non-contrast; low-dose and high-dose CT scans. 131 images are dedicated CTs, and the remaining 9 are the CT component taken from PET-CT exams. This makes the dataset ideal for training and evaluating deep learning algorithms, which ought to perform well in a wide variety of conditions.

To prepare the required training graph dataset, liver meshes are reconstructed from CT scans of 139 patients. To simplify computation, the input slices are first down-sampled with a factor of 0.5 in the plane and spatial resolutions. Next, the 3D surface reconstruction is done using the MarchingCube algorithm, [110] which is a method to extract a surface of interest from a voxel dataset. It is worth noting that patients included in the CT-ORG image dataset were selected based on the presence of lesions in one or more of the labelled organs. The majority of the images in the dataset depict liver lesions, including both benign and malignant cases. As a result, the prepared liver mesh dataset encompasses both normal and abnormal cases. Mesh cardinalities vary from 300 to 3000 points.

3.1.1 Pre-Alignment of Shapes

All structures under consideration need to be rigidly registered with respect to each other, in order to filter out rotation, translation and isometric scaling effects. This is done because the objects may have different orientations in the Euclidean space and therefore their shapes cannot be initially compared. For each dataset, the 3D shapes from all subjects are first aligned using a rigid point set registration described in [65]. Then, the shape coordinates are normalised to the range of $[0, 1]$ for all of the aligned datasets.

3.2 Evaluation Metrics

In general, validation involves evaluating whether a model accurately represents the physical process it intends to depict, and if the model's predictions are comparable to those of the actual system. It is worth mentioning that a model cannot be validated completely, but rather there is a collection of evidence indicating that the model

produces consistent outcomes within a specific range of conditions or parameters [111].

As discussed in Chapter 1, this study follows data-driven approaches to building virtual populations, where virtual patients are generated by sampling from distributions. However, the random generation of individuals does not guarantee that the resulting anatomic case will be physiologically plausible or will belong to the actual population. Therefore, in virtual population modelling, while the model should capture adequate variability within the population, it is necessary to evaluate each member of the model in order to ensure their physiological plausibility. A detailed review in [112] explored methods and findings related to the generation of synthetic medical data. The review, discussing both data fidelity and privacy aspects, provides valuable insights into the challenges and outcomes of creating synthetic medical datasets. Based on this research, we have considered "Visualisation", "Distance metrics" and "Statistical metrics" to evaluate the method and assess the plausibility of generated samples in this study. For the quantitative assessment of the generative models, the performance of generation is evaluated in terms of generalisation, specificity, and clinical relevance metrics.

3.2.1 Generalisation and Specificity

To numerically evaluate the model, two criteria, generalization and specificity, have been proposed in the literature to assess the generation and correspondence quality [33, 113, 114]. The generalisation ability of a generative model indicates the capability of the model to represent unseen samples and thus capture the variability in shapes based on its error when reconstructing unseen actual test data [113]. Specifically, unseen test shapes are first reconstructed using the trained shape models investigated. Subsequently, distance metrics are evaluated between the actual and reconstructed test shapes [113].

The specificity of a model quantifies the quality of the newly generated instances when compared with the training data. A distance is measured between each generated sample in the virtual population and the closest/most similar shape in the actual population. This measure is useful to quantify how similar the randomly generated samples are to the actual samples in the training data.

To quantify the generalisation and specificity errors, the data is split into training and test subsamples. Three distance measures Hausdorff distance (HD), minimum Euclidean distance (ED) and its symmetric distance (ED^*) are considered to report

generalisation and specificity [33].

Hausdorff distance (HD) is measured to determine how different the two shapes g_k and g'_k are: $HD(g_k, g'_k) = \max(\max_{\mathbf{p} \in g_k} \min_{\mathbf{q} \in g'_k} \|\mathbf{p} - \mathbf{q}\|_2, \max_{\mathbf{q} \in g'_k} \min_{\mathbf{p} \in g_k} \|\mathbf{p} - \mathbf{q}\|_2)$. In addition, the average of minimum Euclidean distance between two sets g_k and g'_k is measured as $ED(g_k, g'_k) = \frac{1}{N_k} \sum_{\mathbf{p}} \min_{\mathbf{q} \in g'_k} \|\mathbf{p} - \mathbf{q}\|_2$, and its symmetric distance $ED^*(g_k, g'_k) = ED(g'_k, g_k)$. Distances ED/ED^* measure how two graphs are similar on the average basis but are not suitable for detecting differences in the details of g_k and g'_k , while Hausdorff distance is a lightweight yet nuanced similarity measure to assess the closeness of two graphs.

3.2.2 Clinical Relevance

Inspired by [93], this section presents acceptance criteria for assessing the clinical relevance of the virtual cohorts synthesised using the models investigated in this study.

Given the actual cohort, the acceptance rate \mathcal{A} determines the percentage of synthetic samples in virtual cohorts with biomarkers (e.g. volume indices), within a confidence interval of the distribution of the biomarkers indices observed in the actual population. Specifically, the clinical acceptance rate \mathcal{A} is used as an additional metric to assess the anatomical plausibility of virtual synthesised cohorts. The motivation behind this is to maintain the clinically significant volumetric indices in the synthesized cohorts, with respect to the reference (actual) population.

To compute the acceptance rate \mathcal{A} , three different confidence intervals are considered. First, $[\min, \max]$ interval presents the range of the observed biomarkers in the actual population and we refer to the associated acceptance function as \mathcal{A}^r .

As a second acceptance criterion, a sort of outlier rejection is conducted based on the dispersion observed in the original cohort. More precisely, intervals that accumulate 95% of the probability of finding each biomarker are defined. A Chebyshev's inequality is applied when the actual distribution of the different biomarkers is unknown. Assuming unimodality, Chebyshev's inequality establishes the 95% confidence interval $M \pm 3B$ are defined, based on the corresponding mean (μ), standard deviation (σ) and the mode (M) observed in the actual population. ¹ Here, $B = \sqrt{\sigma^2 + (M - \mu)^2}$ measures the

¹The same interval contains, at least, 91% of the probability density of the distribution if unimodality cannot be assumed.

variability across the data and the acceptance function for these criteria is referred to as \mathcal{A}^M .

In the third acceptance criterion, a normal distribution of the biomarkers is assumed. Dealing with real-world populations, it is reasonable to assume that some of the biomarkers have a Gaussian distribution. Relying on Chebyshev's inequality, the confidence interval $\mu \pm 2\sigma$ is defined where about 95% of the samples lie within two standard deviations from the mean and denote the corresponding acceptance function \mathcal{A}^μ .

Table 3.1 presents the values of the statistics for the biomarkers proposed in LV and liver datasets and the resulting intervals defined by the three acceptance criteria.

3.3 Baseline Generative Shape Model

This section aims at presenting a baseline shape generation model. A method for automatic shape generation of anatomy structures of interest, using statistical models of shape. Due to this, the first step towards the construction of SSMs is establishing point-wise correspondence (i.e. shape matching) across a training set of shapes. Following the registration of a group of shapes using any of the methods discussed in the previous chapter, the estimated posterior probabilities are employed to establish soft correspondences, which in turn are used to train SSMs by the baseline model. More specifically, the baseline generative model introduced in this chapter combines a Registration-based Shape Matching procedure with a PCA-based SSM model, referred to as RSMP. The process of the RSMP generation framework involves two steps: (1) presenting shape matching (i.e. establishing correspondences) and structural normalisation, and (2) SSM-fitting. The former is first used to align training shapes to an assumed template shape and establish correspondences. The estimated correspondences are subsequently projected to training shapes and present structurally normalized regressed shapes.

Here, the basic theory of Coherent Point Drift point-set registration [65] is first presented and then a normalization procedure is considered by regression of shapes on the template model. A brief introduction to the fundamental theory underlying the proposed PCA-based statistical shape models is subsequently provided. Furthermore, the assessment of shape quality presented by the baseline model is demonstrated.

Table 3.1: Statistical description of the biomarkers measured on LV and liver datasets.

Biomarker	μ	M	σ	B	[min, max]	$M \pm 3B$	$\mu \pm 2\sigma$
LV volume	170.18	170.02	6.34	6.34	[133.90, 200.84]	[150.99, 189.04]	[157.50, 182.86]
Liver volume	1073.59	1059.69	238.36	238.77	[266.78, 1361.81]	[343.39, 1775.98]	[596.87, 1550.31]

3.3.1 Point-Set Registration: Coherent Point Drift

Coherent point drift (CPD) is a widely used pair-wise point set registration technique, capable of recovering rigid, affine, and non-rigid transformations. In this chapter, this method is utilised to address the difficult problem of establishing point-wise correspondences (i.e. shape matching) by estimating rigid registration (rigid spatial transformations).

In point cloud rigid registration, the goal is to align two or more point clouds by applying transformations such as translation, rotation, and scaling. Rigid registration assumes that the point clouds represent objects that undergo rigid transformations, meaning the shape of the object remains unchanged while only its position and orientation change. Thus, by employing rigid registration methods for shape matching, the model establishes point-wise correspondences while maintaining the variability of shapes and preventing deformation throughout the registration process. This is crucial in shape matching and generative modelling as it ensures that the shape of the object remains unaltered, which is essential for achieving accurate and meaningful results.

The key aspect of the CPD algorithm is that it considers the problem of point-set registration, one of probability density estimation, where one point-set, represented by Gaussian mixture model (GMM) components, is fitted to the second point-set (considered as data points) by likelihood maximisation, using the Expectation-Maximisation (EM) algorithm for optimisation. The GMM posterior probability for a given data point is maximised when two point sets are aligned and the correspondence is obtained.

The GMM probability density function is

$$p(\mathbf{x}) = \sum_{m=1}^{M+1} p(\mathbf{y}_m)p(\mathbf{x}|\mathbf{y}_m), \quad (3.1)$$

where the points in \mathbf{Y} represent the GMM centroids, and those in \mathbf{X} represent the data points generated by GMM. In an EM-based probabilistic pair-wise registration method, it is assumed that the data points defining the fixed shape (i.e. point set) $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ are independent and identically distributed (i.i.d) and that a noisy observation is derived from a Gaussian distribution centered at a point within the moving point set $\mathbf{Y} = \{\mathbf{y}_m\}_{m=1}^M$. The correspondence probability between data points \mathbf{x}_n and \mathbf{y}_m defined as the posterior probability of the GMM centroid given the data point: $p(\mathbf{y}_m|\mathbf{x}_n) = p(\mathbf{y}_m)p(\mathbf{x}_n|\mathbf{y}_m)/p(\mathbf{x}_n)$.

Considering, equal weighted components and an additional uniform distribution

3.3 Baseline Generative Shape Model

term $p(\mathbf{x}|M+1) = \frac{1}{N}$ to account for noise and outliers in the point-sets, the probability density function (pdf) for a GMM is represented by:

$$p(\mathbf{x}_n|\mathbf{Y}, \theta) = \frac{w}{N} + (1-w) \sum_{m=1}^M \frac{p(\mathbf{x}_n|\mathbf{y}_m, \theta)}{M}, \quad (3.2)$$

where $w, 0 \leq w \leq 1$ denotes the weight of the uniform distribution; N the number of points in the fixed point set and M the number of points in the moving point set (or number of GMM components), also

$$p(\mathbf{x}_n|\mathbf{y}_m, \theta) = \frac{1}{(2\pi\sigma^2)^{D/2}} \exp^{-\frac{\|\mathbf{x}-\mathbf{T}(\mathbf{y}_m, \theta)\|^2}{2\sigma^2}}, \quad (3.3)$$

illustrates the likelihood probability of a point \mathbf{x} sampled from GMM components centred at $\mathbf{T}\mathbf{y}_m$. D is the dimensionality of point sets, \mathbf{T} denotes the desired spatial transformation which can be rigid, similar, affine or non-rigid. θ represents the set of all model parameters, including transformations \mathbf{T} , centroid positions \mathbf{y}_m and variance σ^2 .

Under the i.i.d assumption, the log-likelihood function for all data points can be expressed as a product of the individual conditional probability functions $p(\mathbf{x}_n|\mathbf{y}_m, \theta)$;

$$p(\mathbb{X}|\mathbf{Y}, \theta) = \sum_{n=1}^N \ln(p(\mathbf{x}_n|\mathbf{Y}, \theta)). \quad (3.4)$$

Registration parameters can be estimated by maximising the log-likelihood of the observed data points \mathbf{x}_n with respect to the unknown parameters θ , (using gradient-based optimisation techniques). Alternatively, the EM algorithm may be used to estimate registration parameters θ by minimising the expectation of negative log-likelihood function $Q(\theta^i|\theta^{i-1})$, as described by Equation 3.6. Therefore, in this case, analytical solutions exist for updating estimates of the model and transformation parameters, at each EM-iteration. Throughout this section the current iteration is denoted as i .

$$\theta^i = \underset{\theta}{\operatorname{argmax}} p(\mathbb{X}|\mathbf{Y}, \theta) \equiv \underset{\theta}{\operatorname{argmin}} Q(\theta^i|\theta^{i-1}) \quad (3.5)$$

where

$$Q(\theta^i|\theta^{i-1}) = - \sum_{n=1}^N \sum_{m=1}^M p(\mathbf{y}_m|\mathbf{x}_n, \theta^{i-1}) \ln(p(\mathbf{x}_n|\mathbf{y}_m, \theta^i)) \quad (3.6)$$

Q also serves as an upper bound for the negative log-likelihood function, which is minimised instead of the latter due to its intractable solution. In the EM algorithm, the

minimisation of Q is achieved by iteratively alternating between E-step and M-steps until the convergence. In the E-step, the previous iteration's estimate for the model's parameters θ^{i-1} , are used to compute the posterior probabilities $p(\mathbf{y}_m|\mathbf{x}_n)$, which represent the probability of correspondences between mixture components centred at \mathbf{y}_m and point set \mathbf{x}_n . These posterior probabilities are computed using Bayes' theorem as:

$$\begin{aligned} \text{E-step : } p^{i-1}(\mathbf{y}_m|\mathbf{x}_n) &= \frac{p(\mathbf{y}_m)p(\mathbf{x}_n|\mathbf{y}_m)}{p(\mathbf{x}_n)} \\ &= \frac{\exp\left[-\frac{1}{2}\left\|\frac{\mathbf{x}_n - \mathbf{T}(\mathbf{y}_m, \theta^{i-1})}{\sigma^2}\right\|^2\right]}{\sum_{m=1}^M \exp\left[-\frac{1}{2}\left\|\frac{\mathbf{x}_n - \mathbf{T}(\mathbf{y}_m, \theta^{i-1})}{\sigma^2}\right\|^2\right] + (2\pi\sigma^2)^{D/2} \frac{wM}{(1-w)N}} \end{aligned} \quad (3.7)$$

where, for rigid point set registration, $\mathbf{T}(\mathbf{y}_m; \theta) = s\mathbf{R}\mathbf{y}_m + \mathbf{t}$; $\theta = \{s, \mathbf{R}, \mathbf{t}\}$

$$\begin{aligned} \text{M-step : } \theta^i &= \underset{\theta}{\operatorname{argmin}} Q(\theta^i|\theta^{i-1}) \\ &= \underset{\theta}{\operatorname{argmin}} \frac{1}{2\sigma^2} \sum_{n=1}^N \sum_{m=1}^M p(\mathbf{y}_m|\mathbf{x}_n, \theta^{i-1}) \|\mathbf{x}_n - \mathbf{T}(\mathbf{y}_m, \theta)\|^2 \end{aligned} \quad (3.8)$$

$$+ \frac{D}{2} \ln(\sigma^2) \sum_{n=1}^N \sum_{m=1}^M p(\mathbf{y}_m|\mathbf{x}_n, \theta^{i-1}) \quad (3.9)$$

In summary, the E-step computes the posterior probabilities of the GMM components, the M-step updates estimates for the model and transformation parameters, based on the computed posterior probabilities. The algorithm iteratively alternates between these two steps until a convergence criterion is reached. Presenting shapes as point clouds, posterior probabilities in Equation 3.7 give the correspondence between shapes and template shape.

3.3.2 Statistical Shape Models

In Section 2.3.2 PCA-based Statistical Shape Models are discussed to generate synthetic shapes (represented as point clouds). The technique of PCA is explained here to describe the statistical variations of shapes [31]. The mean shape of this set of normalized aligned shapes is calculated, and modes of shape variation are computed using PCA. PCA reduces data by geometrically projecting them onto lower dimensions called principal components, intending to find the best summary of the data using a limited number of components.

Having a dataset with a consistent number of points becomes a crucial aspect in training SSMs using PCA. It is vital to maintain a uniform number of data points across the samples to enable effective analysis of shape variations and the construction of a meaningful model through PCA. Consequently, to achieve this consistency and enhance the success of SSM training with PCA, a structural normalisation process is considered, which facilitates the creation of structurally normalised training data based on point-to-point correspondences.

For a training set of K shapes ($k = 1, \dots, K$), each shape described by N_k points. The geometric features in shape k are denoted by $\mathbf{x}_k \in \mathbb{R}^{N_k \times 3}$. Regarding the CPD method, the normalized aligned shapes \mathbf{h} are estimated as demonstrated in Equation 3.10, using the correspondence map P (i.e. posterior probabilities derived from Equation 3.7). The estimation of P occurs after aligning each shape to the template model (with N points), and T denotes the corresponding similarity transformation.

$$\mathbf{h}_j = \frac{\sum_{i=1}^{N_k} P_{ji}^T \mathbf{T} \mathbf{x}_i}{\sum_{i=1}^{N_k} P_{ji}}, \quad j = 1, \dots, N \quad (3.10)$$

It subsequently enables the creation of a structurally normalised training dataset, essential for training SSMs through PCA.

Therefore, to illustrate PCA-based SSM, a training set of K shapes is used, represented as structurally normalized point clouds \mathbf{h}_k , where the shapes have a uniform number of N points. The geometric features of the point j in shape k are denoted by $\mathbf{h}_{kj} \in \mathbb{R}^3$, $\mathbf{h}_{kj} = (x_{kj}, y_{kj}, z_{kj})$ with $j = 1, \dots, N$ and $k = 1, \dots, K$.

Using rigid CPD registration, all nuisance pose parameters (such as translation, rotation, and scaling) have been removed. Due to the specific class of shapes covered by this set, the presence of inter-point correlation will consistently be observed. Thus, shape representations could account for point correlations. A reduction in dimensionality could be achieved if some point movements were correlated.

More specifically, let $\{\mathbf{h}_k\}_{k=1}^K$ be the set of all training shapes in matrix form, we are interested in the mean shape $\bar{\mathbf{h}}$ and covariance matrix \mathbf{C} of \mathbf{h} is calculated as follows:

$$g_\mu := \bar{\mathbf{h}} = \frac{1}{K} \sum_{k=1}^K \mathbf{h}_k \quad (3.11)$$

$$\mathbf{C} = \frac{1}{K-1} \sum_{k=1}^K (\mathbf{h}_k - \bar{\mathbf{h}})(\mathbf{h}_k - \bar{\mathbf{h}})^T \quad (3.12)$$

In summary, the shape covariance is represented in a low-dimensional space. The PCA of the shape produces L eigenvectors $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_L]$, and corresponding eigenvalues (i.e. modes of shape variations) $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_L)$ of the covariance matrix computed via Singular Value Decomposition. Hence, assuming the shape class follows a multi-dimensional Gaussian probability distribution, the synthetic shape g^{syn} can be approximated from the following linear generative model:

$$g^{syn} := \mathbf{h} \approx \bar{\mathbf{h}} + \mathbf{U}\mathbf{b} \quad (3.13)$$

$$\mathbf{b} = \mathbf{U}^T(\mathbf{h} - \bar{\mathbf{h}}) \quad (3.14)$$

where vector $\mathbf{b} \in \mathbb{R}^L$ represents the set of parameters used to generate variations in shape, and restricted to $|\mathbf{b}_i| \leq \beta\sqrt{\lambda_i}$; we set $\beta = 3$ to capture 99.7% of shape variability.

3.4 Experimental Results

The shape matching procedure using point set registration-based methods is discussed in Section 3.3.1. By using a Registration-based Shape Matching (RSM) method, which represents shapes as point sets, point-to-point correspondence is calculated by aligning all moving point sets to a fixed one using Equation 3.7. Then it is followed by the domain transformation using Equation 3.10.

Figure 3.1 visualises shape matching results obtained using registration-based method [65] for some examples of LV and liver datasets. In this figure, the number of points for template shapes was 1093 and 1025 for LV and livers respectively.

The findings show that there are certain details that are not captured in the indicated landmarks. This is because the registration is not naturally a learning-based method, it computes a transformation while encouraging the displacement vectors to point in similar directions. Thus, to find a dense correspondence between two shapes (represented as point sets), matching can be performed with only a subsample of points and can be applied on all points. Moreover, the evaluation of the obtained shapes in the template domain compared to the actual shape is conducted for test shapes, employing two distance measures: Hausdorff distance (HD) and Chamfer distance (CD).

The accuracy of shapes obtained by RSM in the fixed domain is summarised in Table 3.2. The high standard deviation values observed for the liver indicate a lack

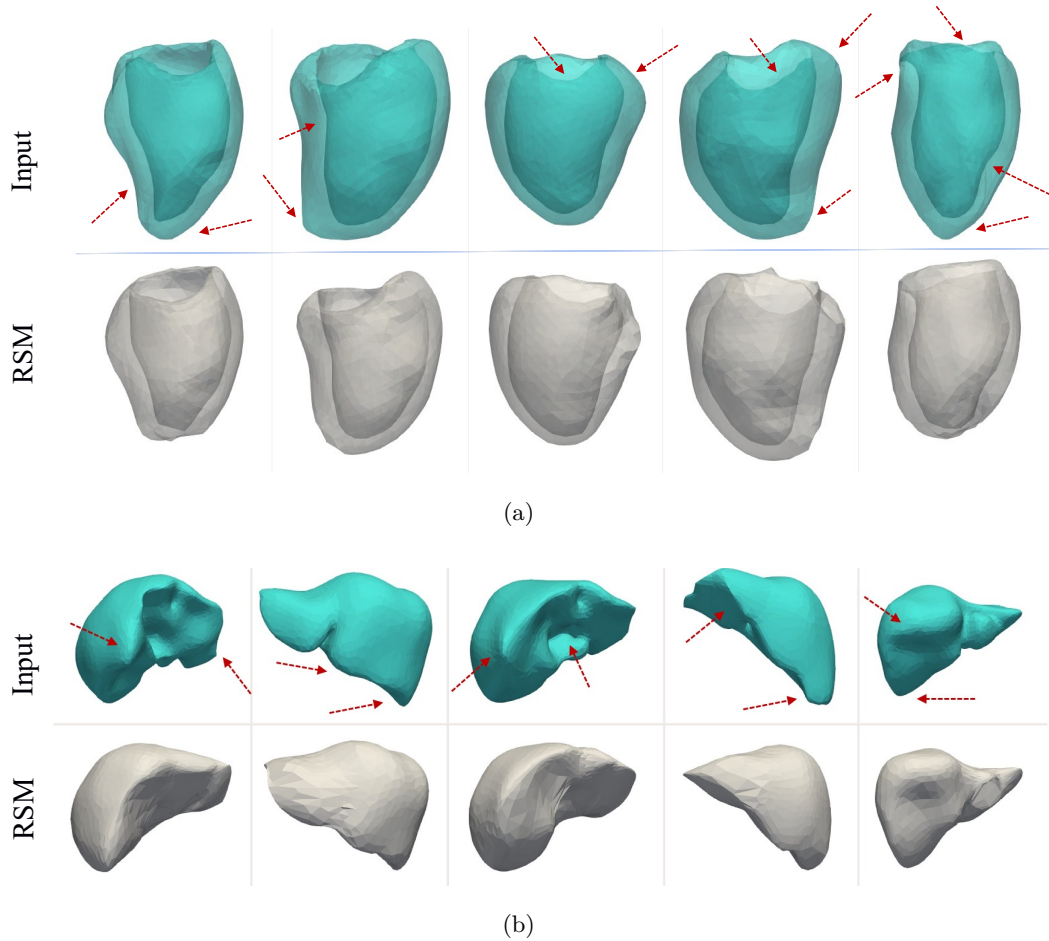


Figure 3.1: Visualisation of some shape matching results obtained by registration-based methods described in Section 3.3.1. Some landmarks are highlighted with red arrows. (a) Cyan-coloured meshes present input LV shapes with cardinality 1586, 1539, 1455, 1150 and 1039 respectively from left to right. Grey-coloured shapes are the normalised meshes with a cardinality of 1093. (b) Cyan-coloured meshes present input liver shapes with cardinality 1532, 1338, 1278, 1030 and 1365 respectively from left to right. Grey-coloured shapes are the normalised meshes with a cardinality of 1025.

of robustness in capturing complex shape variabilities across different subjects. This implies that the liver shape model generated by the RSM method may not effectively handle the intricate variations present in liver shapes among individuals. The reason for this is that these methods tend to enforce rigid transformations uniformly across the entire shape, neglecting the local variations and shape intricacies that may be crucial

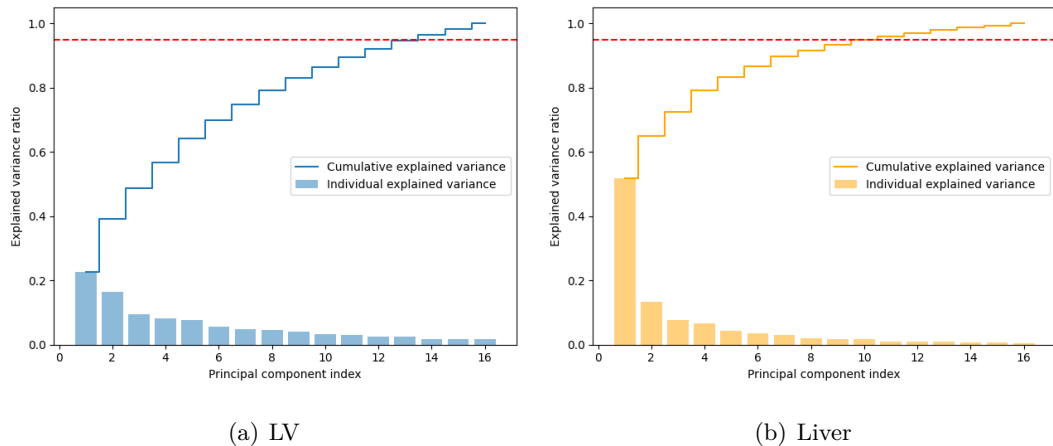


Figure 3.2: The Cumulative Explained Variance (CEV) and the shape variation explained by each of the principal components of the feature vector in the space defined by the PCA, in order of importance. (a) The first 13 modes explain 95% of the total LV shape variation. (b) The first 10 modes explain 95% of the total liver shape variation.

for establishing accurate point-to-point correspondences.

In statistical shape modelling, in order to obtain a set of representative shapes, the selected modes must be able to explain a large percentage of the total shape variation. The cumulative explained variance (CEV) is the relative cumulative frequency of the eigenvalues sorted in descending order.

$$\text{CEV}(l) = \frac{\sum_{i=1}^l \lambda_i}{\sum_{i=1}^L \lambda_i}; \quad 1 \leq l \leq L \quad (3.15)$$

where L is the total number of modes and $\text{CEV}(l)$ gives the cumulative explained variance up to the l -th mode.

Figure 3.2 shows the variance associated with each of the principal components (i.e.

Table 3.2: Shape matching quality assessment for RSM method using two distance metrics HD and CD (mean \pm std) in $[mm]$.

	LV	Liver
HD	8.11 ± 2.13	35.09 ± 14.55
CD	12.04 ± 2.63	256.46 ± 211.57

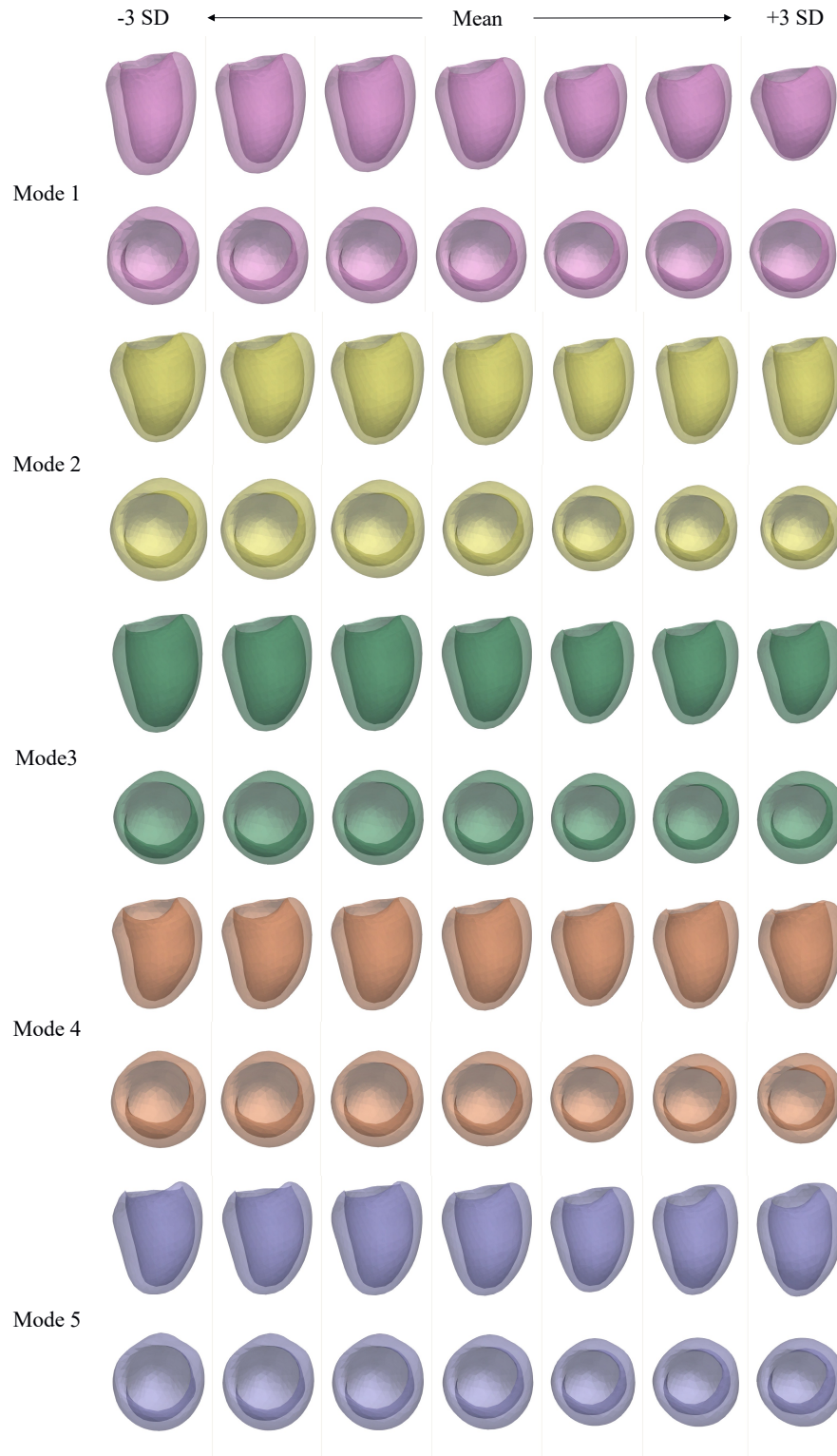


Figure 3.3: Representation of the mean ± 3 standard deviation (SD) of the first five modes of variation in the 3D shape models of left-ventricle.

modes), and the CEV by considering the first $l = 16$ features of the PCA for LV and liver shapes. The first 13 and 10 variation modes can explain 95% of the anatomical variability in the observed samples LV and liver respectively.

The modes of variation in PCA for shapes provide insights into the most prominent and distinct patterns of shape variability within the datasets. As an example, Figures 3.3 and 3.4 visualise the variation of LV and liver shapes respectively, in the first five modes and presents the significant patterns of shape variability observed within the dataset. These modes capture the primary directions or axes along which shapes exhibit the most substantial changes.

In Figure 3.3, the first five modes of variation in PCA for Left Ventricle (LV) shapes present the following changes: The first mode of variation captures the overall size of the LV shape, indicating that it is the most prominent source of variation in the dataset. The second mode captures contraction in specific regions of the LV, such as changes in the width of the ventricular chamber.

The third mode of variation reveals additional shape variations that include changes in curvature, bulging, or concavity in specific regions of the LV, such as alterations in the shape of the ventricular walls. The fourth mode of variation captures shape changes that are independent of the previous three modes. These changes correspond to specific asymmetries or localized deformations within the LV shape, such as uneven thickening of the ventricular walls or variations in the shape of the apex. The fifth mode of variation represents further independent shape changes that are distinct from the previous four modes. These changes include alterations in the overall size or volume of the LV, rotation, or variations in specific regions that deviate from the mean shape in a unique manner.

Further in Figure 3.4, variations of the mean shape in the first five dominant modes by ± 3 Standard Deviation (SD) are shown for the liver. The first mode of variation captures changes in liver size, where positive values indicate smaller and negative values represent larger liver shapes. The second mode of variation illustrates the shape changes that are most prominent after accounting for the mean shape. It can be seen that it presents variations in the length or height of the liver shape. The third mode of variation reveals shape changes related to the curvature or bulging of specific regions within the liver. In the fourth mode, the liver shape changes due to specific asymmetries or deformations. And the fifth mode of variation captures additional shape changes and

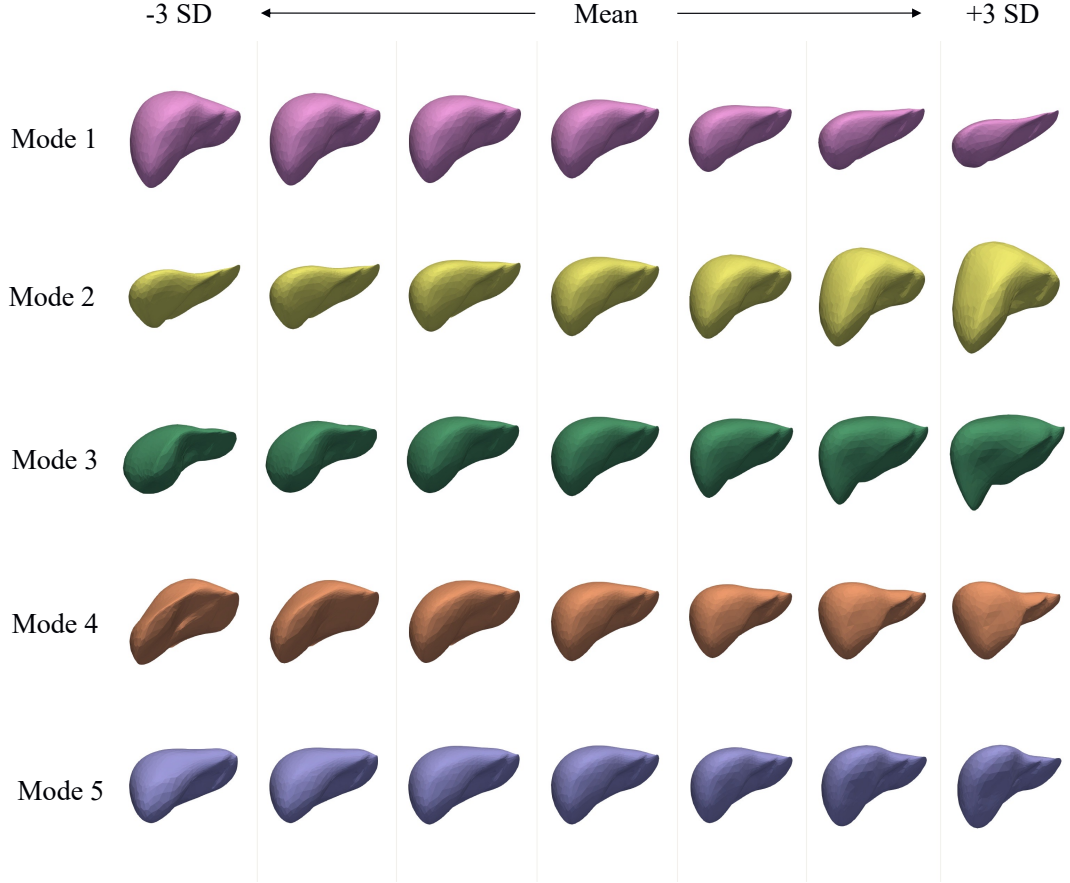


Figure 3.4: Representation of the mean ± 3 standard deviation (SD) of the first five modes of variation in the 3D shape models of the liver.

morphological features that are independent and distinguishable from the preceding four modes.

To synthesise virtual cohorts using the RSMP model, the PCA-based shape model generates samples by uniformly sampling coordinates in the low-dimensional principal sub-space within the defined interval of $[-3\lambda_i, +3\lambda_i]$, λ_i represents the eigenvalue of the i -th principal component, corresponding to the eigenvector spanning the principal sub-space. Figure 3.5 displays the assessment results of the RSMP generative shape framework with uniform sampling (RSMP(Uni)) based on the generalisation and specificity criteria across three different distances. Generalisation ability provides an assessment of the variability in shape that is captured by the principal sub-space representations. Specificity focuses on determining the anatomical plausibility of the synthesized vir-

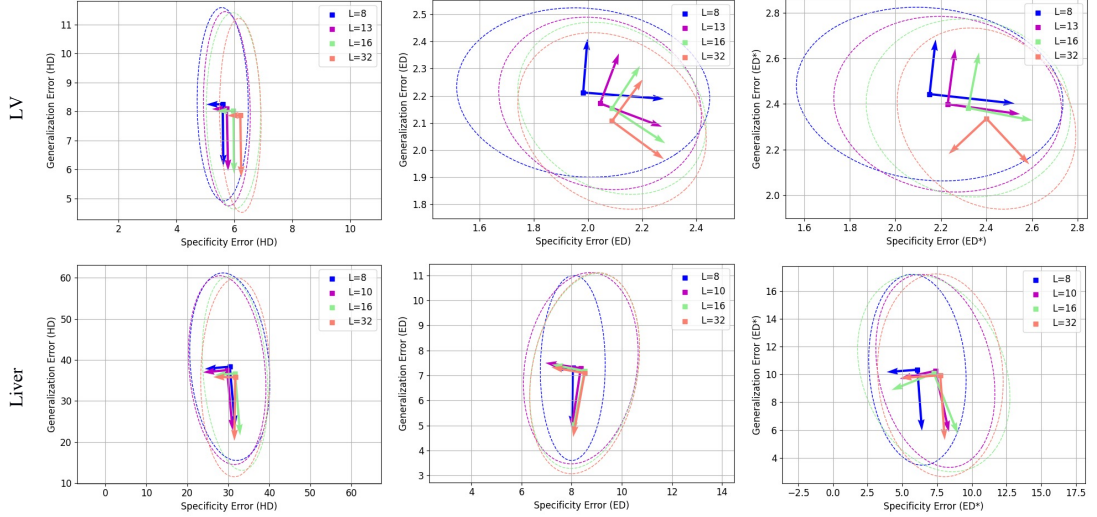


Figure 3.5: Generalisation and Specificity errors (in [mm]) of the RSMP(Uni) model with different principal components L . The first and second rows illustrate results on LV and liver datasets, respectively. The plots show the performance of the models with L values smaller than 32. In each row, HD , ED and ED^* distances are reported from left to right. For each model, the markers, dotted lines and arrows indicate the associated average, confidence ellipse, and the direction of the maximum variability (in error values for test and synthetic data) respectively, for training with various numbers of components.

tual shape cohorts. It measures how well the generated shapes align with realistic anatomical structures.

As shown in Figure 3.5, the observed variations in errors (indicated by arrows) demonstrate that, in most cases, both generalization and specificity errors decrease. This indicates that the model tends to achieve high performance. The figure illustrates that with an increasing number of principal components, there is an increase in the captured variability in shape (i.e. lower generalisation errors) while it indicates minimal changes in specificity errors across the majority of planes. This implies that the RSMP(Uni) generative model can produce realistic shapes for any number of components, while also capturing a greater range of variability with higher numbers of components.

In order to quantify the clinical relevance of synthesised shape populations (defined

Table 3.3: Clinical acceptance rates \mathcal{A} [in %] achieved by RSMP(Uni) generative model for LV and liver volumes.

		RSMP(Uni)
LV	$\mathcal{A}^r : \mathcal{A}_{[\min, \max]}$	93.95
	$\mathcal{A}^M : \mathcal{A}_{M \pm 3B}$	67.15
	$\mathcal{A}^\mu : \mathcal{A}_{\mu \pm 2\sigma}$	47.75
Liver	$\mathcal{A}^r : \mathcal{A}_{[\min, \max]}$	67.63
	$\mathcal{A}^M : \mathcal{A}_{M \pm 3B}$	70.50
	$\mathcal{A}^\mu : \mathcal{A}_{\mu \pm 2\sigma}$	37.41

in Section 3.2), Table 3.3 presents the efficiency of RSMP(Uni) method as assessed through different acceptance criteria. The statistical shape model RSMP assumed that rigidly aligned point sets lie within a Euclidean space and establish point-to-point correspondences. The hypothesized linearity of the model allows for using causal PCA and has proven to be a pragmatic solution for many shape matching tasks. However, in the presence of large shape variations (like liver samples), the non-linearity of the shape space demands more sophisticated analyses. As a result, for datasets with higher morphological features, these methods may generate virtual samples with low clinical relevance.

3.5 Discussion and Conclusion

This chapter introduces the RSMP baseline generative model, which represents shapes as point sets. The model is a PCA-based statistical shape model that establishes point-to-point correspondences using registration-based shape matching methods.

Point set rigid registration methods facilitated the establishment of point-to-point correspondences between shapes. Rigid registration methods, while widely used in aligning shapes, have inherent limitations when it comes to establishing point-to-point correspondences. These methods primarily rely on rigid transformations involving translations, rotations, and scalings. However, they often struggle to capture non-linear deformations or shape variations beyond the scope of rigid transformations.

The main issue with rigid registration methods lies in their assumption of global shape similarity, neglecting local variations that may occur within the shapes. Due to

this oversimplification, these methods may fail to accurately establish correspondences between corresponding points in shapes with complex deformations. Moreover, rigid registration methods do not possess the flexibility to account for shape variability that can arise from factors such as articulation, non-rigid deformations, or anatomical differences among subjects. These methods tend to enforce rigid transformations uniformly across the entire shape, neglecting the local variations and shape intricacies that may be crucial for establishing accurate point-to-point correspondences (shown in Figure 3.1).

Moreover, PCA-based SSM models have limitations in capturing complex variations due to multiple factors, as demonstrated in the obtained results shown in the Figure 3.3.

- **Linearity assumption:** PCA-based SSMs assume that shape variations can be modeled as linear combinations of a set of principal components. This linearity assumption may not hold in cases where shape variations exhibit non-linear or more intricate patterns. Consequently, PCA-based models may not be able to fully capture and represent the complex non-linear shape variations present in the data.
- **Limited representational capacity:** PCA-based SSMs are limited by the number of principal components used to represent shape variations. In practice, only a subset of principal components that capture the most dominant variations is typically utilised due to computational constraints and the need for dimensionality reduction. As a result, PCA-based models may not capture fine-grained or subtle variations that require a larger number of components to accurately represent.
- **Inability to handle out-of-sample variations:** PCA-based SSMs are constructed based on a training dataset, and their performance may deteriorate when applied to shapes that exhibit variations not well-represented in the training data. These models rely on the assumption that the training data sufficiently covers the range of shape variations encountered in the real world. However, if the shape variations in the test or unseen data significantly differ from those in the training data, the PCA-based model may fail to accurately capture and represent these out-of-sample variations.
- **Shape correlations:** PCA-based SSMs assume that shape variations are uncorrel-

ated, meaning that each principal component captures independent shape variations. However, in reality, shapes often exhibit correlated variations, where changes in one part of the shape are associated with changes in other parts. PCA-based models may struggle to capture such correlated shape variations, leading to limitations in their ability to represent complex shape relationships and dependencies.

In general, it is important to address these limitations to ensure that shape models accurately capture the full range of shape variations, enabling more reliable and precise analysis and interpretation of shapes in various applications such as shape generation. To overcome these limitations, deep learning-based approaches for shape matching and generation have been proposed in Chapter 4 and Chapter 5. These methods aim to achieve two goals: first, to capture complex deformations and enable more accurate point-to-point correspondences. Second, to capture more intricate and non-linear shape variations, offering improved performance and representation capabilities compared to PCA-based SSMS (e.g. RSMP baseline model) in capturing complex shape variability. It excels at capturing complex shape variability while maintaining plausible shapes.

CHAPTER 4

A Geometric Deep Learning Framework for
Generation of Shape as Graphs Using Graph
Convolution and Attention Mechanisms:
Leveraging Geometry and Surface Features

4.1 Introduction

In this chapter, a probabilistic approach for generating virtual shape populations from the inconsistent dataset is proposed. The lack of consistent tensor-like representation across the training graphs poses significant methodological challenges which are addressed here. This is achieved through an unsupervised deep learning method embedded in Attention-based Shape Matching and Generation networks (ASMG). A shape matching network derives a learnable set of correspondences using graph embedding techniques, employing a synergy of graph convolutional and attention networks. This is followed by a graph normalisation process over a population and atlas shape learning in an EM-like framework. Subsequently, a generative network learns a probability density function from a set of structurally normalised graphs in the 3D space. Figure 4.1 illustrates the flowchart of the proposed generative model in this chapter.

In the process of developing or selecting generative shape models for specific applications, and sampling strategies for the synthesis of virtual cohorts of anatomy from the former, it is common to make design choices that maintain a balance/trade-off between the variability in shape captured (relative to target/real patient populations) and the anatomical plausibility of the instances synthesised, in the virtual cohorts. In-silico trials require balancing shape variability and anatomical plausibility in synthesised virtual cohorts, as cohorts with high variability may contain unrealistic shapes that are unrepresentative of native anatomy. Therefore, it is essential to generate a plausible variant of data while local detailed information on the shape is preserved. This leads to the introduction of models that realistically describe the anatomy and its variation in the population.

This chapter advocates synthesising cohorts in a manner that maximizes variability and plausibility in synthesised anatomical shapes, and therefore,

- It is imperative to model accurate and realistic shapes in the attention-based shape matching (GCN-ATT) network. To this end, in this chapter, two variants of the GCN-ATT are presented: (1) where representations of vertices are only defined by geometric features (i.e. spatial positions of points); and (2) hybrid representations of vertices defined by spatial positions of points, with the associated vertex normal vectors. The former is of particular interest in this chapter (henceforth referred to as sGCN-ATT), while the latter serves to highlight the capability of the presented work for preserving more morphological details and

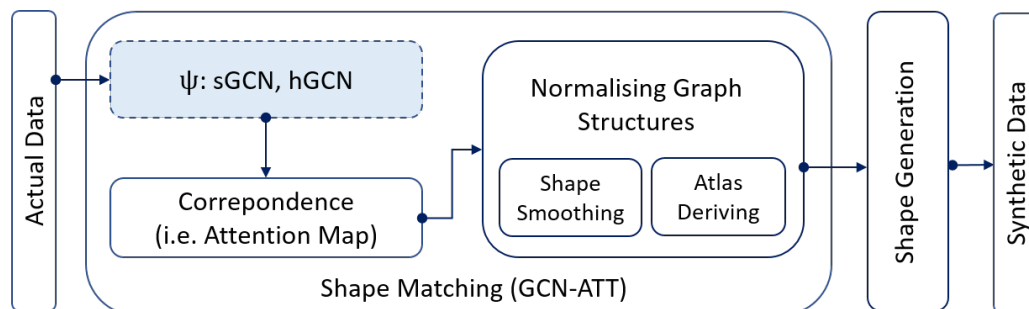


Figure 4.1: Schematic showing our geometric deep learning framework for shape matching and generation. Different-sized actual shapes first go through the shape matching phase, in which, the network derives a learnable set of correspondences and learns population-derived atlas shape, then flows into the generation network, and finally produces the realistic synthetic 3D shapes.

variation (referred to as hGCN-ATT).

- Based on the study in [93] that explored the benefits and drawbacks of different sampling strategies, a Gaussian sampling strategy is employed in a generative network in order to generate virtual anatomical shape chimeras.

4.2 ASMG Model

Problem Statement: Given a set of observed surface meshes, our goal is to develop a shape generative model using a graph representation.¹ Let $g_k = (V_k, E_k)$ be the k -th graph specified by a set of its vertices and edges, with $|V_k| = N_k$ vertices and E_k be the set of edges, connecting the vertices. The sparse adjacency matrix $\mathbf{A}_k \in \{0, 1\}^{N_k \times N_k}$ represents the edge connections. The training will be assumed to be a set of graphs $G = \{g_k\}_{k=1}^K$. Assume that the features of the vertex v in graph k are denoted by $\mathbf{x}_{kv} \in \mathbb{R}^{d_x}$ and that the matrix $\mathbf{X}_k \in \mathbb{R}^{N_k \times d_x}$ aggregates all of the node feature vectors for the graph k . It is further assumed that in the sGCN-ATT network, the position of each vertex in a mesh is characterized by spatial features (i.e. $d_x = 3$); while in the hGCN-ATT network, the vertex representation is denoted by the concatenation of the 3D vertex coordinate and surface normal vectors (i.e. $d_x = 6$). The corresponding

¹The term "graph" will be used to refer to shapes/meshes later on in this study.

Table 4.1: Commonly used notations.

Notations	Descriptions
K	Number of observations.
k	Observed graph.
μ	Atlas (i.e. template) graph.
g	A graph.
g'	A smoothed normalised graph.
V	The set of nodes in a graph.
v	A node $v \in V$.
E	The set of edges in a graph.
e_{ij}	An edge $e_{ij} \in E$.
\mathcal{N}_v	The neighbors of a node v .
\mathbf{A}	The graph adjacency matrix.
N	The number of nodes, $N = V $.
d_x	The dimension of a node feature vector.
d_z	The dimension of a latent node feature vector.
$\mathbf{X} \in \mathbb{R}^{N \times d_x}$	The feature matrix of a graph.
$\mathbf{x}_v \in \mathbb{R}^{d_x}$	The feature vector of the node v .
\mathbf{x}_{kv}	The feature vector of the node v in the k -th graph.
$\mathbf{Z} \in \mathbb{R}^{N \times d_z}$	The latent feature matrix of a graph.
$\mathbf{z}_v \in \mathbb{R}^{d_z}$	The latent feature vector of the node v .
$\mathbf{h} \in \mathbb{R}^{N \times d_h}$	The feature matrix of a normalised graph.
$\mathbf{h}' \in \mathbb{R}^{N \times d_h}$	The feature matrix of a smoothed normalised graph.
\mathbf{h}_{kv}	The feature vector of the node v in the k -th normalised graph.
\mathbf{h}'_{kv}	The feature vector of the node v in the k -th smoothed normalised graph.
$\Phi_k \in [0, 1]^{N_\mu \times N_k}$	Vertex-wise correspondence (i.e. attention map) between graph g_k and g_μ .
$\mathbf{Z}' \in \mathbb{R}^L$	The latent vector of a smoothed normalised graph.

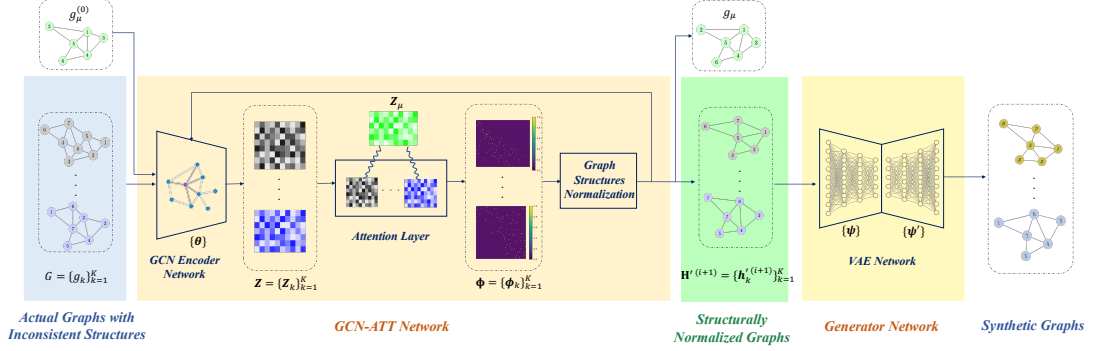


Figure 4.2: Overview of the proposed ASMG framework for graph generation from a graph population with different graph structures.

stochastic latent variable of v is also introduced and denoted as $\mathbf{z}_{kv} \in \mathbb{R}^{d_z}$, summarised in a matrix $\mathbf{Z}_k \in \mathbb{R}^{N_k \times d_z}$. Unless particularly specified, the notations used in this chapter are illustrated in Table 4.1.

Figure 4.2 shows a diagram of the proposed method, where a hierarchical framework is trained to generate shapes by minimizing the following cost function

$$L(\mathbf{H}', \mathbf{X}_\mu, \boldsymbol{\psi}, \boldsymbol{\psi}') = L_1(\mathbf{H}', \mathbf{X}_\mu) + L_2(\boldsymbol{\psi}, \boldsymbol{\psi}'). \quad (4.1)$$

First, a graph convolutional encoder network (GCN) (parameterised by $\{\boldsymbol{\theta}\}$) computes the nodal embedding vectors in the latent space. These embedding vectors are used to compute the attention weights and the pair-wise correspondences across the vertices of the training and template graphs. The template graph model initiates with a canonical shape and iteratively updates as will be explained in Section 4.2.1.

This results in normalising the structure of input meshes such that each normalised g_k is now represented as the k^{th} column of a matrix $\mathbf{H} = (\mathbf{h}_k)_{k=1}^K$. Given the latter, L_1 (refers to shape matching) is minimised to refine the corresponding smoothed shapes (stored as columns of \mathbf{H}'), and the atlas μ , which normalises the structures of the input meshes. Next, L_2 (refers to generation) is minimised to train a variational auto-encoder (parameterised by $\{\boldsymbol{\psi}, \boldsymbol{\psi}'\}$) using the structure normalised graphs. The overall cost is iteratively minimised in L_1 in each step until convergence is achieved (see Algorithm 1). Algorithm 1 is described in the general context. It can be employed for both s/hGCN-ATT settings.

In this study, a correspondence map is employed for structural normalization, specifically to transform a graph from the actual domain to the atlas domain. This trans-

Algorithm 1 atlas shape learning and establishing correspondences

```

 $g_\mu$ .Initialise()
 $\mathbf{Z}_k \leftarrow GCN_\theta.Eval(); \forall g_k \in G$ 
while  $error \geq 0.001$  do
   $\mathbf{Z}_\mu \leftarrow GCN_\theta.Eval()$ 
   $\Phi_k \leftarrow Attention(\mathbf{Z}_k, \mathbf{Z}_\mu)$ 
   $\mathbf{h}_k \leftarrow \Phi_k \mathbf{X}_k$ 
   $\mathbf{h}'_k.Initialise() \leftarrow \mathbf{h}_k$ 
  while  $error \geq 0.001$  do
     $\mathbf{h}'_k \leftarrow \langle \partial L_{11}, \partial \mathbf{h}'_k \rangle = 0$ 
  end while
   $\mathbf{X}_\mu \leftarrow \langle \partial L_{12}, \partial \mathbf{X}_\mu \rangle = 0$ 
end while

```

formation involves transferring the vertex graph positions from the actual domain to the atlas domain. This process is applied in both settings 1 and 2, where the framework establishes a correspondence map from different graph representations (either spatial or hybrid).

In the rest of the study, the atlas graph is indicated as g_μ , and its matrix representation is denoted by \mathbf{X}_μ . In the following, each loss term is explained in more detail.

4.2.1 Attention-based Shape Matching (ASM)

The purpose of this section is to develop a novel learning-based shape matching approach that leverages graph neural networks and attention mechanisms. More specifically, as illustrated in Figure 4.2, the correspondences between graphs are computed in the latent space using a GCN followed by an attention module (named GCN-ATT network), and the graph structures are normalised across the population.

Presenting the 3D surface mesh as a graph, this matching procedure aims to determine correspondence based on the similarity measure of local node embeddings without requiring to solve an optimization problem during inference. To compute nodal embeddings, graph neural networks enable us to efficiently and flexibly aggregate information through graph nodes and edges, which generate a powerful representation of shapes.

Once the graph embedding is learned, an attention mechanism is applied to those embedded features to learn correspondence across the population as illustrated in Figure 4.3.

The process is a forward-backwards regression of graph datasets on the template model (i.e. atlas graph) in an EM-like framework. This method finds simultaneously the atlas shape and the correspondence functions to warp this atlas into each one of the observed shapes. Indeed, this algorithm iteratively updates the vertex-to-vertex correspondences and template graph model using node embedded information in the point clouds \mathbf{Z}_k .

This method builds a template graph model, that is representative of a graph ensemble and vertex-to-vertex correspondences between the template and each training graph. The framework allows us to structurally normalise the given training graph dataset with different connectivities on a unique template graph, g_μ , with cardinality $|V_\mu| = N_\mu$.

4.2.1.1 Computing the vertex-wise correspondences

The attention mechanism is first introduced by Vaswani *et.al* in [103]. Attention mechanisms are generally employed to encourage the models to focus on certain parts of input data [115]. Hence, in this study, a soft attention mechanism is incorporated into the matching procedure to determine where and what shape matching should focus on. The model is designed to identify essential features for learning informative representations of graphs in Graph Convolutional Networks (GCN). Additionally, it computes significant relationships, such as vertex-to-vertex correspondences, between the learned representations by measuring the similarity of local node embeddings (as illustrated in Figure 4.3). Therefore, the attention mechanism guides the latent embeddings \mathbf{Z} to be conducted in a more concentrated way and thus generates a more accurate correspondence map. To achieve this, first, the latent embeddings of nodes $\mathbf{Z}_k = \Psi_\theta(\mathbf{X}_k, \mathbf{A}_k)$ and $\mathbf{Z}_\mu = \Psi_\theta(\mathbf{X}_\mu, \mathbf{A}_\mu)$, are computed by a shared graph neural network Ψ_θ for the observed graph g_k and the atlas graph g_μ , respectively. The embeddings are normalised using $\mathbf{Z}_\mu = \left(\mathbf{z}_i^{(\mu)} / |\mathbf{z}_i^{(\mu)}|\right)_{i=1}^{N_\mu}$ and $\mathbf{Z}_k = \left(\mathbf{z}_i^{(k)} / |\mathbf{z}_i^{(k)}|\right)_{i=1}^{N_k}$. Next, for every g_k , the soft correspondence (i.e. soft attention maps) is obtained, as the mapping function in the embedded-space paradigm

$$\Phi_k = \text{Softmax}(\lambda \mathbf{Z}_\mu \mathbf{Z}_k^T), \quad (4.2)$$

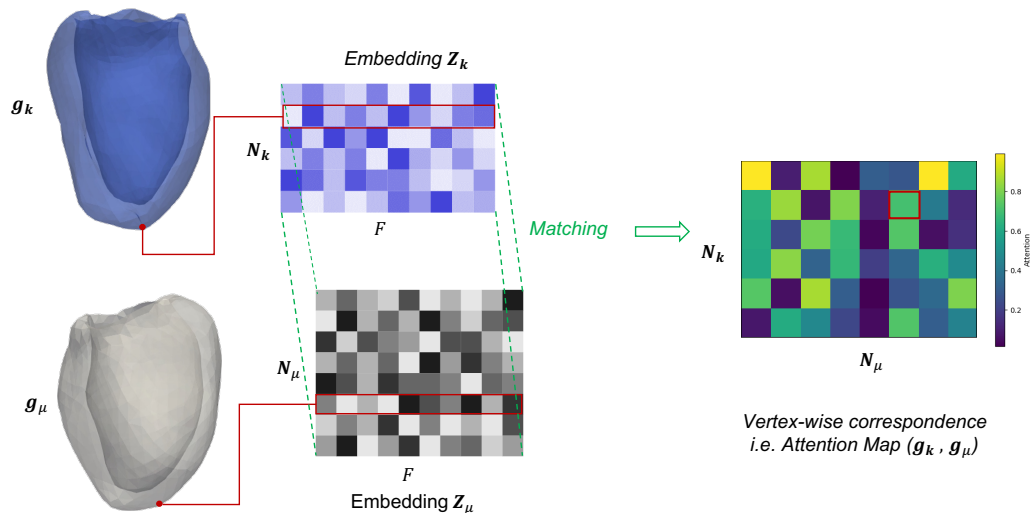


Figure 4.3: Illustration of the proposed attention-based shape matching procedure which finds the vertex-wise correspondence between two shapes by measuring similarities between the nodal embeddings.

where $\Phi_k \in [0, 1]^{N_\mu \times N_k}$ and hyper-parameter λ is inverse variance of $\{\mathbf{Z}_\mu \mathbf{Z}_k^T\}_{k=1}^K$. The attention structure in Equation 4.2 allows discovering (dis)similarities between the template graph and input instance in the latent space. A detailed description of graph convolutional neural network Ψ is given in Section 4.2.1.3.

4.2.1.2 Normalising Graph Structures (Regression on Atlas)

The soft correspondence matrix is a map from the node function space $\mathcal{F}(\mathbb{R}^{N_k})$ to node function space $\mathcal{F}(\mathbb{R}^{N_\mu})$ thus allows for domain transformations. Having the estimated vertex-to-vertex correspondences in $\Phi = \{\Phi_k\}_{k=1}^K$, for every input graph g_k , a structurally normalised graph(/shape) is represented as $\mathbf{h}_k \in \mathbb{R}^{N_\mu \times d_h}$ using a soft attention mechanism ($\mathbf{h}_k := \phi_k \mathbf{X}_k$). Soft attention is implemented to focus more on the relevant features of the shape. In soft attention, irrelevant areas are discredited by multiplying the corresponding shape features with a low weight. Hence, to obtain shapes in the template domain (i.e to warp the atlas shape to each observed shape), the soft attention mechanism directly passes node geometric features vector \mathbf{x} along with the soft

correspondences

$$\mathbf{h}_{kj} = \sum_{i=1}^{N_k} \phi_{ji} \mathbf{x}_{ki}; j = \{1, \dots, N_\mu\}, \quad (4.3)$$

where \mathbf{h}_{kj} is a d_h dimensional feature vector corresponding to j -th node in the k -th normalised graph and the weights ϕ_{ji} are i -th element of the j -th row on the matrix Φ_k . The structure normalised shapes can be represented as the columns of the matrix \mathbf{H} . Having \mathbf{H} , the cost L_1 can be minimised, which consists of two parts $L_1 = L_{11} + L_{12}$, serving the purposes described below:

Shape smoothing using graph Laplacian: The normalised graphs obtained using Equation 4.3 do not generally result in smooth shapes and may present noisy surfaces. To establish smoother shapes, for every shape (represented by \mathbf{h}_k), a corresponding secondary shape g' , with matrix representation \mathbf{h}'_k is derived, by minimising the following cost function,

$$L_{11}(\mathbf{H}') = \frac{1}{2} \sum_{k=1}^K \sum_{j=1}^{N_\mu} |\mathbf{h}'_{kj} - \mathbf{h}_{kj}|^2 + \frac{\gamma}{2} \sum_{j=1}^{N_\mu} \sum_{q=1}^{N_\mu} a_{jq} |\mathbf{h}'_{kj} - \mathbf{h}'_{kq}|^2. \quad (4.4)$$

This cost function employs a Laplacian term to enhance smoothness where a_{jq} are elements of the adjacency matrix \mathbf{A}_μ , and γ is a hyper-parameter. Equation 4.4 can be iteratively optimised w.r.t. the \mathbf{h}'_{kj} ¹ resulting in

$$\mathbf{h}'_{kj}{}^{(i+1)} \leftarrow \frac{\mathbf{h}_{kj} + \gamma \sum_{q \in \mathcal{N}_{kj}} \mathbf{h}'_{kq}{}^{(i)}}{1 + \gamma \sum_{q \in \mathcal{N}_{kj}} a_{jq}}; \quad \forall j \in N_\mu \quad (4.5)$$

where i indexes iteration steps, and \mathcal{N}_{kj} is the set of neighbours of j -th vertex with cardinality $|\mathcal{N}_{kj}|$ (see appendix A.1 for a full description). This Laplacian smoothing consists in moving every vertex of the mesh towards the average location of its topological adjacent vertices. Applying Equation 4.5 multiple times shrinks the shape slightly but mainly smooths sharp details. In order to maintain shapes as close to their real counterparts as possible, the selected graphs are smoothed using i steps of Laplacian smoothing until they reach $error = \sum_{j=1}^{N_\mu} |\mathbf{h}'_{kj} - \mathbf{h}_{kj}|^2 > 0.001$ (as shown in Algorithm 1). As a result, each smoothed normalised shape g'_k is presented by matrix feature representation $\mathbf{h}'_k \in \mathbb{R}^{N_\mu \times d_h}$. Finally, a generation network in Section 4.2.2 learns a probability density function from a set of $\{g'_k\}_{k=1}^K$ graphs, which allows us to generate cohorts of artificial shapes.

¹A closed-form solution can be obtained as shown in Appendix A.1

Deriving the atlas μ : Having the smooth shape representation in \mathbf{H}' , a matrix representation \mathbf{X}_μ for the atlas graph g_μ is inferred by minimising the following cost which, similar to L_{11} , entails a Laplacian term for smoothness

$$L_{12}(\mathbf{X}_\mu) = \frac{1}{2} \sum_{k=1}^K \sum_{j=1}^{N_\mu} |\mathbf{h}'_{kj} - \mathbf{x}_{\mu j}|^2 + \frac{\gamma}{2} \sum_{j=1}^{N_\mu} \sum_{q=1}^{N_\mu} a_{jq} |\mathbf{x}_{\mu j} - \mathbf{x}_{\mu q}|^2 \quad (4.6)$$

This cost is iteratively optimised with respect to \mathbf{X}_μ .¹ Starting from an initial template shape $\mathbf{X}_\mu^{(0)}$, a new template $\mathbf{X}_\mu^{(i+1)}$ is iteratively computed from $\mathbf{X}_\mu^{(i)}$ according to

$$\mathbf{x}_{\mu j}^{(i+1)} \leftarrow \frac{\sum_k \mathbf{h}'_{kj} + \gamma \sum_{q \in \mathcal{N}_{\mu j}} \mathbf{x}_{\mu q}^{(i)}}{K + \gamma \sum_{q \in \mathcal{N}_{\mu j}} a_{jq}}; \quad \forall j \in N_\mu \quad (4.7)$$

where $\mathbf{x}_{\mu j}$ is a feature vector corresponding to j -th node in the atlas graph and $\gamma = N_\mu / \max(\mathcal{N}_{\mu j})$. The atlas shape g_μ is built with a fixed adjacency matrix \mathbf{A}_μ so that only the vertex positions are optimised and the connectivity of nodes is preserved. $K + 1$ virtually regressed graphs are obtained by computing these correspondences for all irregular input graphs with different structures. Therefore, this method builds an atlas shape/graph model, which is labeled as the template here. Algorithm 1 describes the atlas graph learning and establishing correspondences in our model. L2-norm error is considered for iterative computation in Equation 4.7 and Equation 4.5. The algorithm starts from a canonical shape $\mathbf{X}_\mu^{(0)}$ for the atlas. In subsequent iterations, the atlas is initialized using the updated representation from the previous iteration. This process repeats, with the atlas being refined in each iteration to better capture the anatomical variability in the dataset. This allows the atlas to adapt to the nuances and complexities of the dataset. In general, this iterative process of atlas learning allows the model to adapt to the variability in the dataset, resulting in a representative atlas.

4.2.1.3 Graph Convolutional Neural Network (Ψ)

Section 4.2.1.1 described how embedded features from graph convolutional neural networks contribute to establishing correspondences. Graph embedding refers to the approach of learning latent feature representations for the nodes or edges in a graph. The basic principle is to learn encodings for the nodes in the network such that the similarity in the embedding space reflects the similarity in the graph [116]. A detailed

¹A closed-form solution can be obtained as shown in Appendix A.2

explanation of the proposed graph convolutional neural network Ψ is presented in this section.

In order to capture both local and global structural information, deep neural network Ψ uses a variational graph autoencoder architecture parameterized by $\{\boldsymbol{\theta}, \boldsymbol{\theta}'\}$. Basically, a variational graph autoencoder is a framework for unsupervised learning on graph-structured data based on variational autoencoders [91]. Here, the variational graph autoencoder network Ψ takes the adjacency matrix \mathbf{A} and node features \mathbf{X} as input and tries to recover the feature matrix \mathbf{X} through the hidden layer embeddings \mathbf{Z} (further described in Equation 4.11). This network is trained in an unsupervised manner by maximising the Evidence Lower Bound (ELBO) L_Ψ w.r.t. the variational parameters:

$$L_\Psi(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sum_{k=1}^K \underbrace{(\mathbb{E}_{q_\theta(\mathbf{Z}_k|\mathbf{X}_k, \mathbf{A}_k)}[\log p_{\theta'}(\mathbf{X}_k|\mathbf{Z}_k, \mathbf{A}_k)] - w_{kl})}_{L_{rec}} \underbrace{D_{KL}[q_\theta(\mathbf{Z}_k|\mathbf{X}_k, \mathbf{A}_k) \parallel p(\mathbf{Z}_k)]}_{L_{reg}}. \quad (4.8)$$

This loss function is composed of a "reconstruction term", that tends to make the encoding-decoding scheme as performant as possible, and a "regularisation term", that tends to regularise the organisation of the latent space. Where D_{KL} is the Kullback-Leibler divergence (KL divergence) between the approximate posterior $q_\theta(\mathbf{Z}_k|\mathbf{X}_k, \mathbf{A}_k)$ and the prior distribution $p(\mathbf{Z}_k)$, weighted by w_{kl} . Unit Gaussian distribution defines a prior distribution $p(\mathbf{Z}_k) = \prod_{i=1}^{N_k} N(\mathbf{z}_{ki}; \mathbf{0}, \mathbf{I})$.

The inference model is parameterised by graph convolutional layers as:

$$q_\theta(\mathbf{Z}_k|\mathbf{X}_k, \mathbf{A}_k) = \prod_{i=1}^{N_k} q(\mathbf{z}_{ki}|\mathbf{X}_k, \mathbf{A}_k) \quad (4.9)$$

with $q(\mathbf{z}_{ki}|\mathbf{X}_k, \mathbf{A}_k) = N(\mathbf{z}_{ki}|\mathbf{m}_{ki}, \text{diag}(\boldsymbol{\sigma}_{ki}^2))$

where $\mathbf{m} = \text{GCNlayer}_m(\mathbf{X}, \mathbf{A})$ is the matrix of mean vectors \mathbf{m}_n and $\log \boldsymbol{\sigma} = \text{GCNlayer}_\sigma(\mathbf{X}, \mathbf{A})$. The decoder decodes the latent variables into 3D shapes by

$$p_{\theta'}(\mathbf{X}_k|\mathbf{Z}_k, \mathbf{A}_k) = \prod_{i=1}^{N_k} p(\mathbf{x}_{ki}|\mathbf{Z}_k, \mathbf{A}_k). \quad (4.10)$$

As discussed in Section 2.2.2, the spatial-based approaches in graph convolutions are able to handle inconsistent graph populations, where the convolution is performed in local Euclidean neighbourhoods. While the spectral-based approaches use domain-dependent Fourier bases, so generalisation is limited to inputs with identical graph

connectivity. This problem is avoided by working directly in the spatial domain. Hence, the spatial-based graph convolution layers are considered to build a neural network Ψ capable of dealing with different-sized graph populations which learn a latent space for establishing vertex-wise correspondences.

Spatial-based Feature-Steered graph (FeaSt) convolution proposed in [3] is utilised to construct the network layers and to infer the probability distributions in Equation 4.8. These convolution layers locally perform graph convolutions on each node (the details are discussed in Section 2.2.2.2). That is, the convolution operator learns features from the preceding network layer to dynamically determine the correspondences between filter weights and graph neighbourhood, rather than relying on static pre-defined local pseudo-coordinate systems. For each node v , on graph k , the output of the FeaSt convolution operator defines the latent embedding vectors using

$$\mathbf{z}_{kv} = \mathbf{b} + \frac{1}{|\mathcal{N}_{kv}|} \sum_{n \in \mathcal{N}_{kv}} \sum_{m=1}^M q_m(\mathbf{x}_{kv}, \mathbf{x}_{kn}) \mathbf{W}_m \mathbf{x}_{kn}, \quad (4.11)$$

where \mathbf{b} is a learnable bias and \mathcal{N}_{kv} is the set of neighbours of the v -th vertex, on graph k , with cardinality $|\mathcal{N}_{kv}|$. The convolution operator also dedicates a soft correspondences of $q_m(\mathbf{x}_{kv}, \mathbf{x}_{kn})$ (in Equations 2.22) to learn assignment \mathbf{x}_{kn} to m -th learnable weight matrix \mathbf{W}_m .

In this chapter, the proposed geometric deep learning network Ψ is employed in two variants: **sGCN** and **hGCN**.

Setting 1-sGCN

In the first scenario, the feature representation of shape is only defined by the spatial position of points, that lie in 3D Euclidean space. A feature matrix is denoted by \mathbf{X} whose i -th row is $\mathbf{x}_i = (x_i, y_i, z_i)$ (i.e. $d_x = 3$). Therefore, the embedding network Ψ (named sGCN in this scenario) computes nodal embeddings from the variable-size graphs by minimising ELBO in Equation 4.8, where the reconstruction loss is the squared Euclidean distance (L_2 -norm) between real and reconstructed shapes by the decoder: $L_{rec} = \frac{1}{2} \sum_{k=1}^K \sum_{i=1}^{N_k} \|\mathbf{x}_{ki}^{rec} - \mathbf{x}_{ki}\|^2$. \mathbf{X} and \mathbf{X}^{rec} denote the input shape features and its reconstruction respectively.

Setting 2-hGCN

In more complex anatomical structures, spatial features may not be sufficient to describe the details, therefore more powerful feature representation is achieved by hybrid representation. This hybrid representation in GCN results in the extraction of richer latent features. Hybrid representation is driven by both spatial node positions and their associated normal vectors. First, a brief discussion on vertex normal is presented here:

To leverage surface information in triangular meshes, vertex normals can be used to represent surface features along with geometry features in triangular meshes. Each vertex on the triangle mesh is assigned a normal vector, whose direction is determined by the underlying surface. Vertex normals are traditionally estimated from a local neighbourhood of surrounding face normals using some form of local averaging [117].

The most widespread estimation of the vertex normal which is referred to as Mean Weighted Equally (MWE), is introduced by Gouraud [118] as

$$\mathbf{n}_{MWE} = \frac{\sum_{t=1}^f \mathbf{n}_t}{|\sum_{t=1}^f \mathbf{n}_t|} \quad (4.12)$$

where \mathbf{n}_t is the face normal of triangle t and f is the total number of triangles that share a common vertex for which the vertex normal is to be estimated. However, in this algorithm, resulting normal vectors depend on the meshing of the surface since the normal of each adjacent facet contributes equally to the vertex normal. Consequently, the normals will change if the meshing changes for example through adaptive tessellation of a deforming surface.

In order to obtain a result that depends only on the local vertex geometry and not on the meshing, the spatial contribution of each facet must be considered. Therefore, a vertex normal approximation that incorporates the geometric contribution of each facet (refers as Mean Weighted by Angle (MWA)) is proposed in [119]. In this algorithm, vertex normal is defined as

$$\mathbf{n}_{MWA} = \frac{\sum_{t=1}^f \alpha_t \mathbf{n}_t}{|\sum_{t=1}^f \alpha_t \mathbf{n}_t|} \quad (4.13)$$

where α_t is the angle between two edges e_m and e_{m+1} of a face t sharing the vertex.

In hybrid representation, embedding network Ψ (named hGCN in this scenario) learns combined embeddings of geometry and surface. The feature matrix is a horizontal concatenation of the vertex geometries and vertex normals and represented

by \mathbf{X} where i -th row is $\mathbf{x}_i = (x_i, y_i, z_i, \mathbf{n}_i^x, \mathbf{n}_i^y, \mathbf{n}_i^z)$ (i.e. $d_x = 6$) and normal vectors \mathbf{n} are computed using Equation 4.13. To minimise the loss function in Equation 4.8, $L_{rec} = \frac{1}{2} \sum_{k=1}^K \sum_{i=1}^{N_k} \|\mathbf{x}_{ki}^{rec} - \mathbf{x}_{ki}\|^2$ and the regularisation term consists of D_{KL} and the normal consistency loss; $L_{reg} = w_{kl} D_{KL}[q\theta(\mathbf{Z}_k | \mathbf{X}_k, \mathbf{A}_k) \parallel p(\mathbf{Z}_k)] + w_{norm} \sum_{k=1}^K \sum_{p \in g_k^{rec}} (1 - \|\mathbf{n}_{kp}\|^2)$ where p is the vertex on the k -th reconstructed graph. Further, to regress shapes on the atlas and reconstruct the structurally normalised shape by soft attention mechanism $\mathbf{h}_k := \phi_k \mathbf{X}_k^{1:3}$.

Once the encoder-decoder pair has been trained, the decoder is essentially tossed away, while the encoder acts as a feature extractor for establishing vertex-wise correspondence, associating to each input \mathbf{X} , an embedding matrix \mathbf{Z} .

4.2.2 Generative Modeling

Here, a generation network (designed in β -VAE framework) is trained to derive a pdf from the set of normalised graphs. The probability of node variations is approximated in the latent space $\mathbf{Z}'_k \in \mathbb{R}^L$ via a posterior $q_\psi(\mathbf{Z}'_k | \mathbf{h}'_k)$. The likelihood of the observed population is estimated by drawing samples from this approximate posterior probability. The variational lower bound L_2 is optimised w.r.t the variational parameters ψ, ψ' :

$$L_2(\psi, \psi') = \sum_{k=1}^K (\mathbb{E}_{q_\psi(\mathbf{Z}'_k | \mathbf{h}'_k)} [\log p_{\psi'}(\mathbf{h}'_k | \mathbf{Z}'_k)] - \beta D_{KL}[q_\psi(\mathbf{Z}'_k | \mathbf{h}'_k) \parallel p(\mathbf{Z}'_k)]), \quad (4.14)$$

where \mathbf{h}'_k presents graph geometric features after normalisation process (in Section 4.2.1) and the first term is a reconstruction error which computes the squared Euclidean distance (L2-norm) between input and reconstructed shapes by the decoder. D_{KL} denotes the Kullback-Leibler divergence and computes the divergence between the Gaussian prior $N(0, I)$ and posterior distributions of the latent space \mathbf{Z}' and the KL divergence is weighted by hyperparameter β . Novel shape samples are generated by simply sampling from the prior normal distribution and using the sample as the input to the decoder, which converts that into a random synthetic mesh having the same connectivity as the atlas.

Table 4.2: The detailed information on the generative model ASMG for two different settings, where matrices \mathbf{C} and \mathbf{n} denote vertices locations and vertices normals respectively, \oplus means a concatenation of matrices.

Setting	GCN network Ψ	ATT mechanism
sGCN-ATT-VAE	$\mathbf{X} = \mathbf{C}$ (i.e. $d_x = 3$)	$\mathbf{h} := \phi\mathbf{X}$ (i.e. $d_h = 3$)
hGCN-ATT-VAE	$\mathbf{X} = \mathbf{C} \oplus \mathbf{n}$ (i.e. $d_x = 6$)	$\mathbf{h} := \phi\mathbf{X}^{1:3} = \phi\mathbf{C}$ (i.e. $d_h = 3$)

4.3 Results and Discussion

This section presents the experimental settings and the evaluation of the proposed framework using real image-derived clinical datasets discussed in Chapter 3. Two datasets with a different number of shapes and different complexities in the structure are used to evaluate the model’s versatility and to explore different aspects of the model. The proposed algorithm covers the left-ventricular shapes and complex liver shapes. A large test set of 200 LV cases and a small test set of 28 liver cases are used to assess the generalisability of ASMGs compared to baseline RSMP models, using both limited and large training sets.

The proposed generative model ASMG is investigated in two different settings. A more detailed illustration of these settings can be found in Table 4.2.

Experimental Setup

All experiments are carried out using Python *v.3.8.0*, PyTorch, and PyTorch Geometric [120]. In the training phase, ADAM optimizer [121] is used. The network Ψ uses the spatial-base graph convolution layers with 64, 64, 128, d_z -dim hidden layers for the encoder (which are mirrored for the decoder) with convolutional filter weight matrices of size 8. More specifically, inspired by [3] the spatial-based graph convolution operator (named feature-steered convolutional operator) is used where the operator dynamically assigns filter weights to the node’s neighbourhoods according to the features learned by the network. All the internal layers use batch normalisation and Leaky ReLU as activation layers. The structure of this network is also shown in Table 4.3. The learning rate and w_{kl} are empirically set to $1e^{-3}$, and w_{norm} is $1e^{-2}$. The Laplacian weighting factor γ in Equation 4.4 is empirically set to 50. In addition, the iterative optimization in Equation 4.7 starts from a canonical shape for the atlas. Finally, fully-connected

Table 4.3: The structure of network Ψ , where N denotes the cardinality of input shape, $d_z = 128$ and $d_x = 3$ or 6 for s/hGCN settings.

Layer	Filter weight matrix size	Input size	Output size
Spatial GCN	8	$N \times d_x$	$N \times 64$
Batch Norm	-	$N \times 64$	$N \times 64$
Spatial GCN	8	$N \times 64$	$N \times 64$
Batch Norm	-	$N \times 64$	$N \times 64$
Spatial GCN	8	$N \times 64$	$N \times 128$
Batch Norm	-	$N \times 128$	$N \times d_z$
Spatial GCN (μ)	8	$N \times d_z$	$N \times d_z$
Spatial GCN ($\log\sigma$)	8	$N \times d_z$	$N \times d_z$
Spatial GCN	8	$N \times d_z$	$N \times 128$
Batch Norm	-	$N \times 128$	$N \times 128$
Spatial GCN	8	$N \times 128$	$N \times 64$
Batch Norm	-	$N \times 64$	$N \times 64$
Spatial GCN	8	$N \times 64$	$N \times 64$
Batch Norm	-	$N \times 64$	$N \times 64$
Spatial GCN	8	$N \times 64$	$N \times d_x$

layers are used in the structure of the β -VAE network with the same number of hidden units for the encoder and decoder. Encoder architecture is fully-connected layers of size 512, 256, 768, 128, L , and LeakyReLU is used after every layer. The decoder is a mirrored version of the encoder. In the training phase, the learning rate is set to $lr = 1e^{-3}$, scheduled with stepsize = 100 and gamma = 0.5. The β values for LV and liver datasets are empirically set as $2e^{-6}$ and $2e^{-3}$, respectively. Tuning the parameters, (like regularization weights β and $w_k l$) in the variational autoencoder networks is a crucial step in balancing the reconstruction and regularization terms in the loss function. The β parameter controls the strength of the regularization term, which influences the disentanglement of the learned latent space. As beta increases, the regularization term becomes more dominant in the overall loss. This higher emphasis on regularization encourages the model to learn more disentangled and compact representations in the

latent space.

4.3.1 Evaluation of GCN-ATT Network Performance

In this section, the performance of the ASM shape matching network (i.e., (s/h)GCN-ATT) is evaluated, which involves the establishment of probabilistic vertex-to-vertex correspondences between the constructed template and each training graph, followed by the regression of the input mesh graphs on the atlas and the normalisation of their structure.

Correspondence Quality

Validation of results produced by correspondence algorithms is a very complex problem since ground-truth correspondences are unavailable. The most common validation without using a ground-truth correspondence is a visual assessment of results and only a few criteria have been proposed in the literature to evaluate the correspondence quality [41]. Figures 4.4 and 4.5 present the quality assessment for vertex-to-vertex correspondences of some landmark points (pink) on two examples of LV and liver shapes. The meaningful vertex-to-vertex correspondences between the atlas and each case shape are visualized using blue lines and the colour-coded vertices on the shapes, quantify the soft attention map (i.e. established vertex-to-vertex correspondences). Those vertices in yellow are most likely to correspond to the landmarks. Results demonstrate that the established correspondences by sGCN-ATT network are meaningful for anatomical landmarks on LV shapes (e.g. endocardial, epicardial and LV base and apex) and important anatomical landmarks on liver shapes (e.g. landmarks that visualise the liver as a triangle).

Additional qualitative results on the LV and liver datasets are presented in Figure 4.6 and Figure 4.7, respectively. The colour-coded shapes aid in visually understanding the spatial relationships between points in different shapes and their corresponding vertices. In our method, two scenarios, sGCN-ATT and hGCN-ATT, are used to predict vertex-to-vertex correspondences for each shape, resulting in shape matchings among the collection of shapes. The corresponding points between the atlas (colour-code) and obtained shapes g' are shown with the same colour.

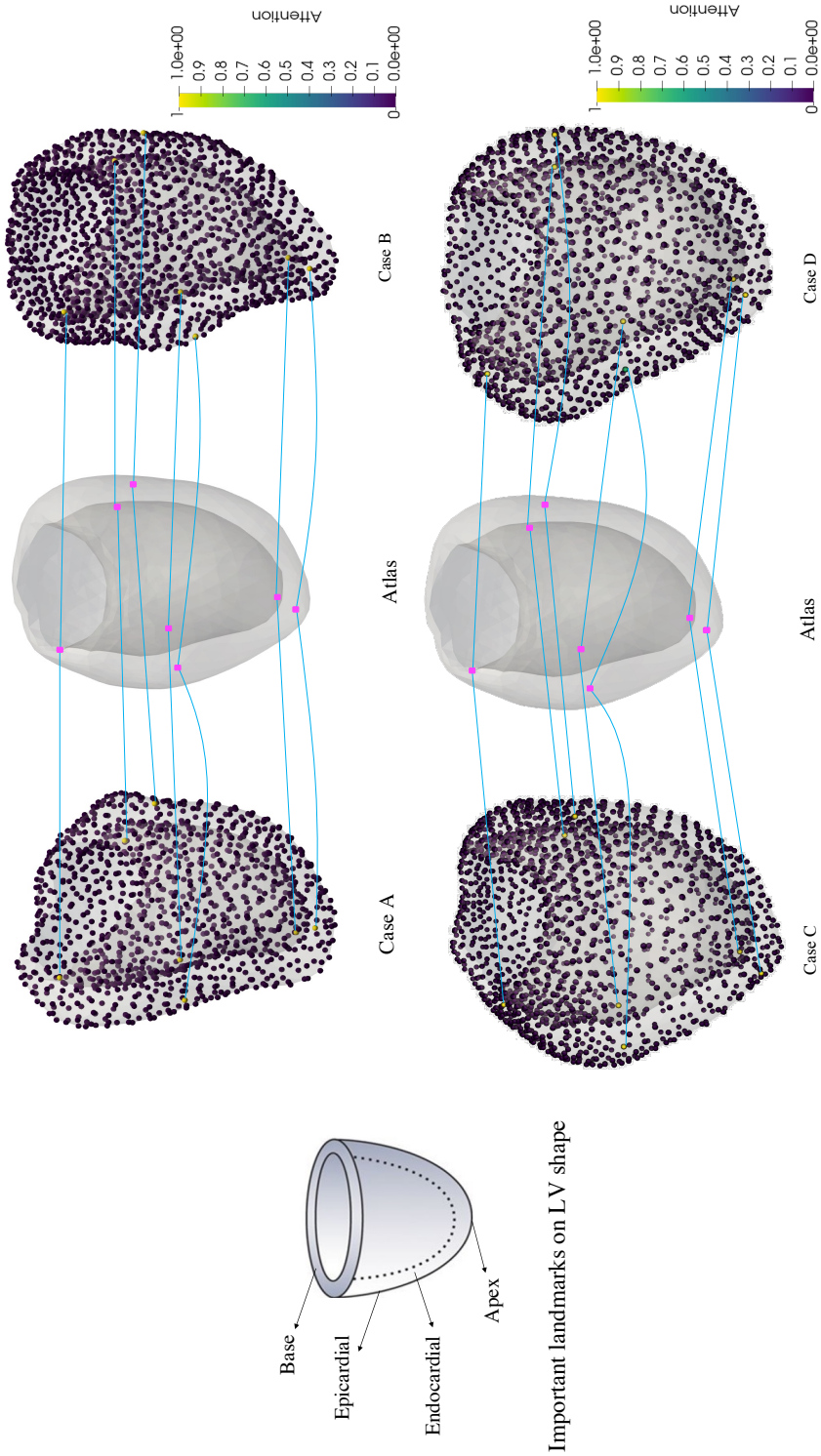


Figure 4.4: Visualisation and Quantification of the established vertex-to-vertex correspondences by sGCN-ATT for four cases of LV shapes. Examples for illustrating meaningful correspondences (blue lines) between a sparse set of landmark points (pink) on an atlas and subjects; color-coded vertices on the shapes present the value of soft attention map $(\phi_{ji} \in [0, 1]; j \in \{\text{landmark points}\}, i = \{1, \dots, N_k\})$.

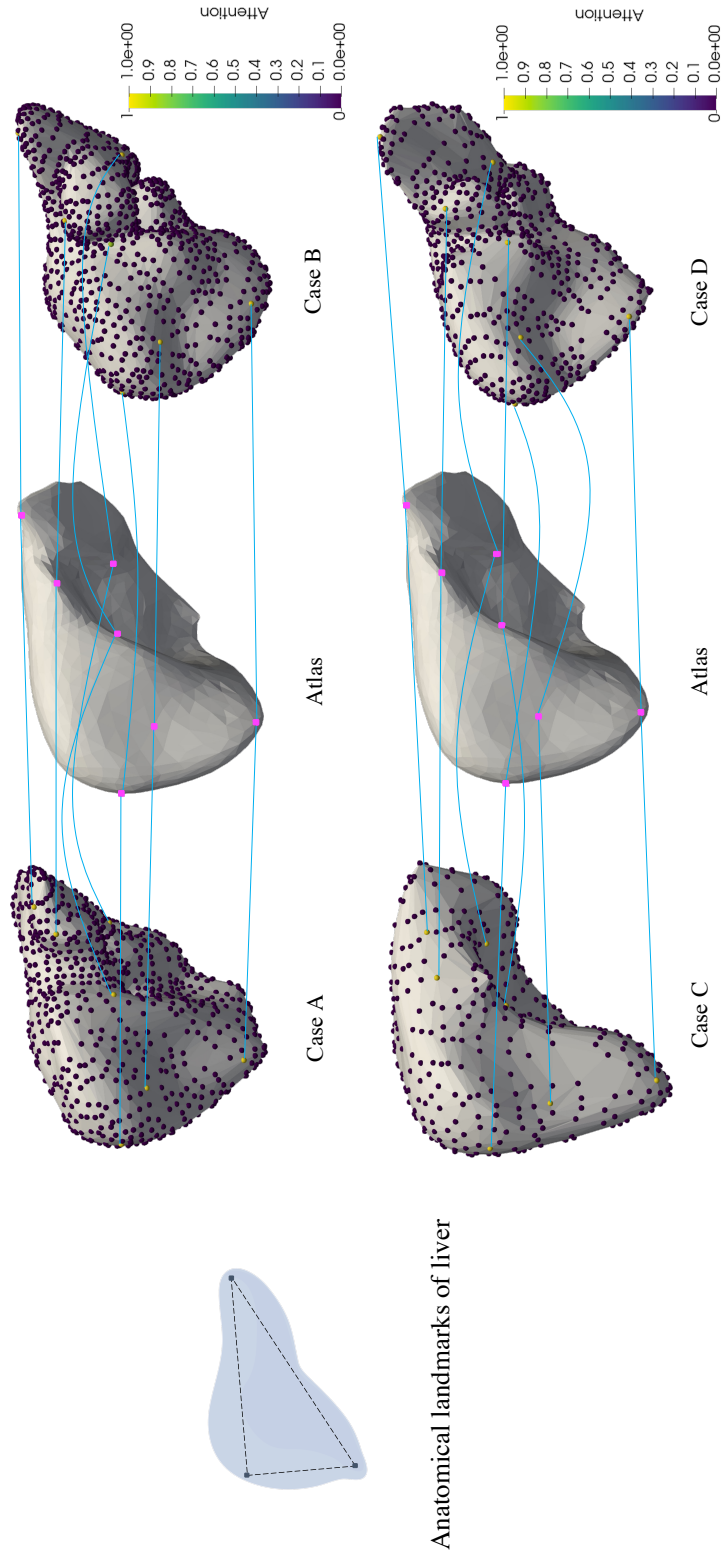


Figure 4.5: Visualisation and Quantification of the established vertex-to-vertex correspondences by sGCN-ATT for four cases of liver shapes. Examples for illustrating meaningful correspondences (blue lines) between a sparse set of landmark points (pink) on an atlas and subjects; color-coded vertices on the shapes present the value of soft attention map ($\phi_{ji} \in [0, 1]; j \in \{\text{landmark points}\}, i = \{1, \dots, N_k\}$).

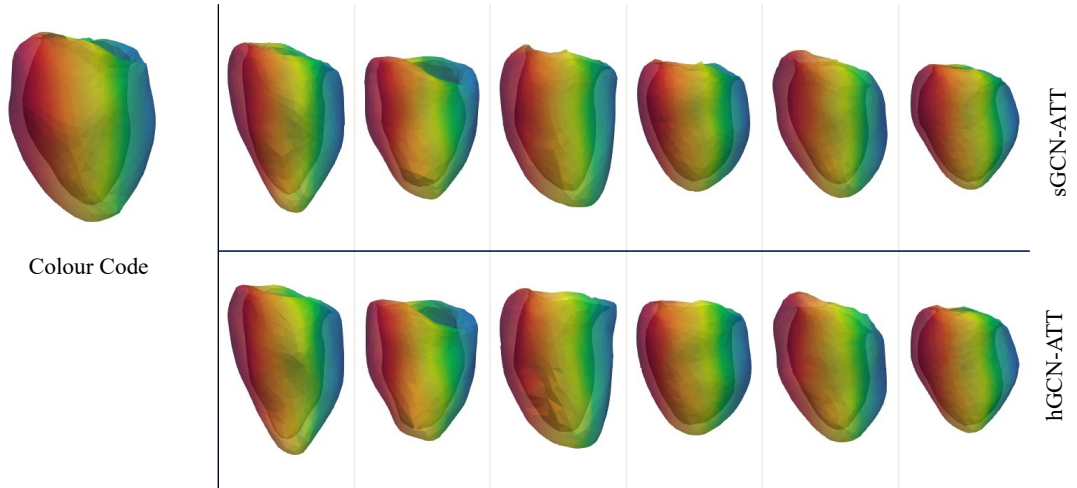


Figure 4.6: Qualitative shape matching results using the ASM method (in s/hGCN-ATT settings) on the LV dataset. The atlas shape is shown merely for visualisation purposes (colour code).

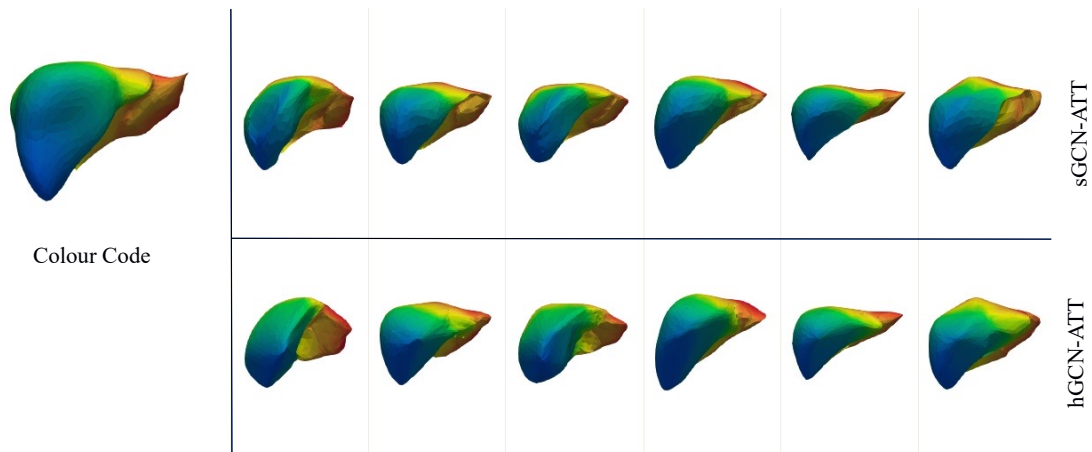


Figure 4.7: Qualitative shape matching results using the ASM method (in s/hGCN-ATT settings) on the liver dataset. The atlas shape is shown merely for visualisation purposes (colour code).

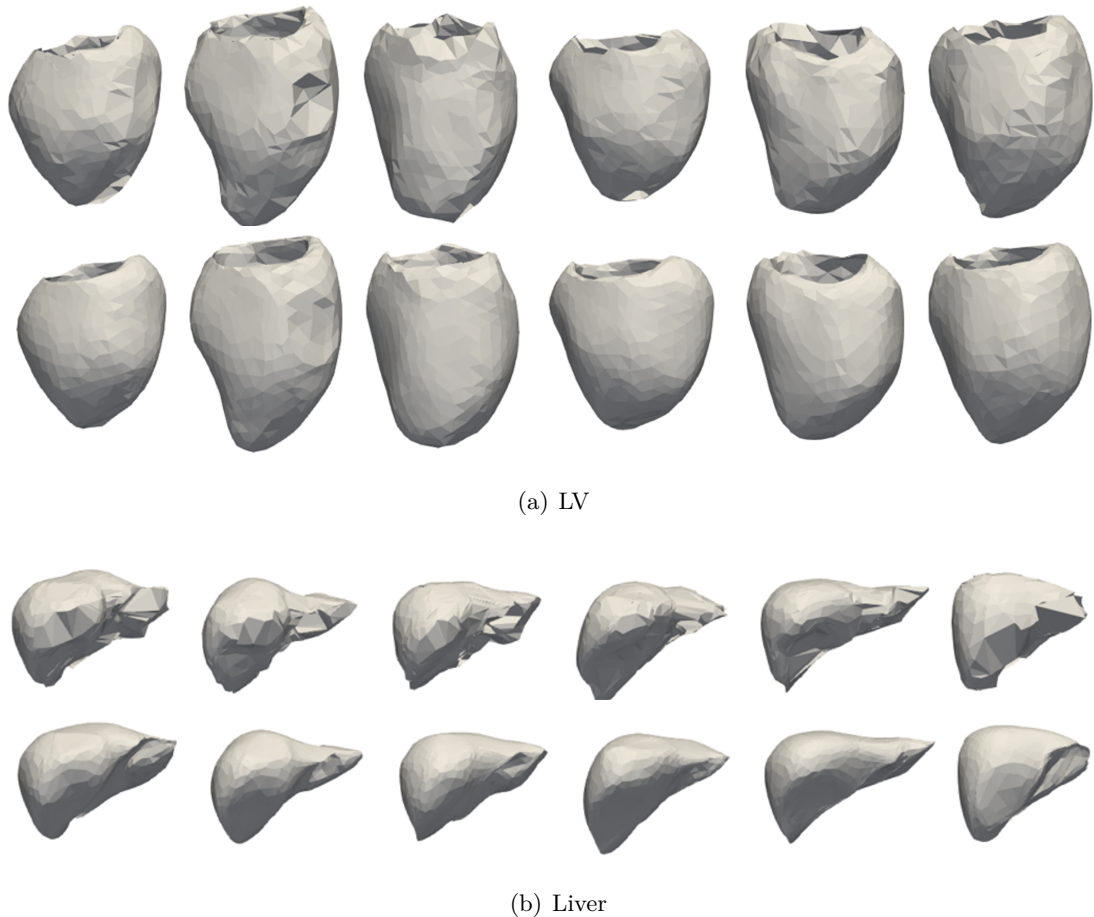


Figure 4.8: Visualization of the shape regressions and graph Laplacian smoothing. The results are presented for the LV and liver samples. In each shape category, the top and bottom present the regressed (\mathbf{h}_k) and smoothed shapes (\mathbf{h}'_k) obtained using Equation (4.3) and Equation (4.5), respectively.

Evaluation of Graph Normalisation (Regression on Atlas)

In this section, the performance of the GCN-ATT network in regressing the input mesh graphs on the atlas and normalising their structure is evaluated.

To demonstrate the efficacy of the Laplacian smoothing on the regressed graphs, some examples of different LV and liver shapes are shown in Figure 4.8. In each category, the first row presents the estimated shapes before the smoothing (i.e. \mathbf{h}_k 's), and the second row shows the corresponding smoothed \mathbf{h}'_k obtained with Equation 4.5. It can be observed that adequate details are preserved while surface imperfections are

largely eliminated in the smoothed meshes. This can be important to improve the overall quality of the generated samples.

The accuracy of the shape matching procedure was evaluated by investigating the proposed attention-based shape matching (ASM) framework in two settings (s/hGCN-ATT) and comparing them with the rigid Registration-based Shape Matching (RSM) method proposed in [65], which is a point set registration method. This comparison highlights the benefits of the graph representation of shapes and the attention mechanisms for the shape matching procedure. Hence, the correspondence maps obtained from different methods RSM and (s/h)GCN-ATT, are utilised to transform actual shapes g_k from the domain \mathbb{R}^{N_k} to \mathbb{R}^{N_μ} . The qualitative results are presented in Figures 4.9 and 4.10 with quantitative results shown in Table 4.4.

For a qualitative evaluation, Figure 4.9 and Figure 4.10 visualise some examples of the network input (g_k) and normalised surface mesh (g'_k) for five sample cases on the LV and liver dataset respectively, obtained from the different methods. Visual inspection of results shows that normalised shapes obtained from our framework (especially with hybrid representation) are more realistic and present meaningful correspondences for anatomical landmarks on LV shapes (e.g. endocardial, epicardial, LV base, and apex) and liver shapes, whereas the results obtained from RSM show some disorders and lack details on both datasets. When comparing normalised shapes in the atlas domain (grey-colored) with cyan-colored shapes in the actual domain, it becomes evident that methods unable to establish accurate correspondences (such as RSM) are unable to effectively present normalized shapes in the atlas domain. Generally, the RSM method struggles to delineate some landmarks on the LV and liver shapes (shown by red arrows).

The quality of obtained shapes in the template domain was further evaluated using two distance metrics: Hausdorff distance (HD) and Chamfer distance (CD). Table 4.4 summarises the accuracy of the test shapes in the template domain obtained by different methods on LV and liver datasets (i.e. the 200 LV and 28 liver cases test set). Obtained results show that our similarity-based method (ASM) maintains high accuracy when compared with the registration-based method. These results suggest that the hybrid representation scenario (i.e hGCN-ATT model) produces more accurate features derived from the GCN network Ψ , thus providing high-quality correspondences. This enhancement is attributed to the hGCN-ATT network’s capability to explicitly

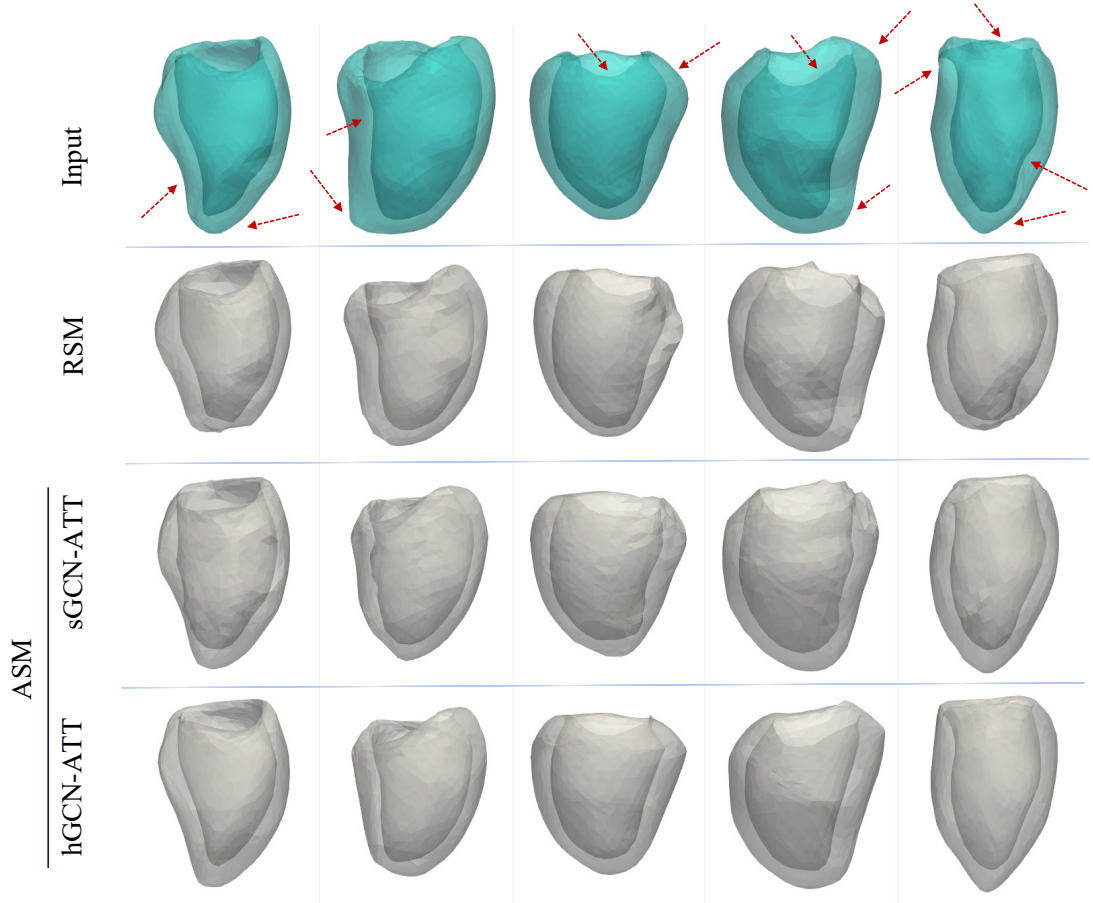


Figure 4.9: Examples show the comparison between different shape matching approaches RSM and ASM (s/hGCN-ATT) for deriving correspondence on the LV dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1586, 1539, 1455, 1150$ and 1039 respectively from left to right. Grey-coloured shapes are the normalised meshes g'_k with cardinality $N_\mu = 1093$ obtained from different methods. Notice endocardial, epicardial, LV base, and apex landmarks. Some of the landmarks are shown by red arrows.

encode surface information by representing shapes as graphs. By achieving lower mean Hausdorff and Chamfer distances, our method demonstrates good normalisation quality in arbitrary target domains, while the low standard deviation values demonstrate its robustness. This is because, representing shapes as graphs, our method followed an efficient spatial-based geometric deep learning strategy and considered a learning-based fully-differentiable shape matching procedure that aimed to reach a data-driven neigh-

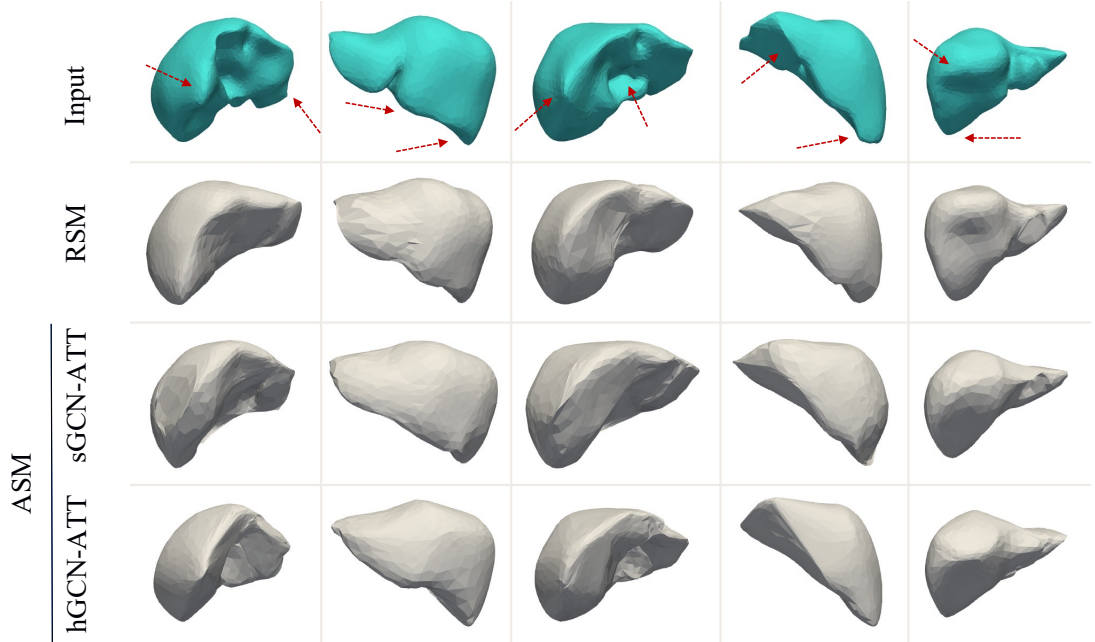


Figure 4.10: Examples show the comparison between different shape matching approaches RSM and ASM (s/hGCN-ATT) for deriving correspondence on the liver dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1532, 1338, 1278, 1030$ and 1365 respectively from left to right. Grey-coloured shapes are the normalised meshes g'_k with cardinality $N_\mu = 1025$ obtained from different methods. Red arrows highlight some landmarks.

Table 4.4: Shape Matching Quality: comparison between different methods using two distance metrics HD and CD (mean \pm std) in $[mm]$. **Bold** values show a significant difference between the methods with a p-value < 0.001 using the statistical paired t-test.

		ASM		
		RSM	sGCN-ATT	hGCN-ATT
LV	HD	8.11 ± 2.13	8.32 ± 1.77	6.43 ± 1.44
	CD	12.04 ± 27.63	9.91 ± 1.54	9.85 ± 0.94
Liver	HD	35.09 ± 14.55	32.79 ± 10.16	27.30 ± 7.71
	CD	254.46 ± 211.57	139.66 ± 82.66	94.75 ± 35.99

bourhood between matched node pairs without the need to solve any optimisation. The RSM model which is based on point cloud representation of shapes, and uses registration for shape matching, under-performed compared to the ASM (s/hGCN-ATT) models generally on both datasets, in line with results from Figures 4.9 and 4.10. It can be concluded that a significant role is played by the graph representation of shapes and the attention mechanisms in the shape matching process.

Impact of Atlas Resolution

In this section, the sensitivity of GCN-ATT network performance to the atlas resolution is evaluated. Figure 4.11 shows the accuracy of regressed shapes on atlases with different resolutions over LV and liver datasets where the Hausdorff distance (HD) is considered to report the accuracy of the shapes after normalising their structures using established vertex-to-vertex correspondences (in the sGCN-ATT setting). It can be seen that low-resolution atlas shapes negatively impact the accuracy and reliability of the shape normalisation process. For the LV dataset, $N_\mu = 1093$ is the high-performance model. For the more complex liver dataset, the resolution is further increased. The results indicate a preference for a higher resolution atlas for the graph normalisation process, and HD is roughly constant between $N_\mu = 1025$ and $N_\mu = 4125$. However, $N_\mu = 1025$ is the simplest model which presents an accurate description for liver shapes.

The performance slightly degrades at the highest resolution for LV shapes, but this is not observed for liver shapes. This can be attributed to the specific anatomical characteristics, shape variability in the test dataset, and dataset properties of these organs. Additionally, in some cases, higher-resolution atlas shapes may introduce noise or ambiguity due to smaller anatomical details that are not consistent across different LV shapes. This can lead to errors in the shape regression(/structure normalisation) process.

A population-derived atlas (with the optimised resolution) within our deep learning framework (i.e. s/hGCN-ATT) is illustrated in Figure 4.12 on both LV and liver datasets. As expected, the constructed atlas by hGCN-ATT network presents more details.

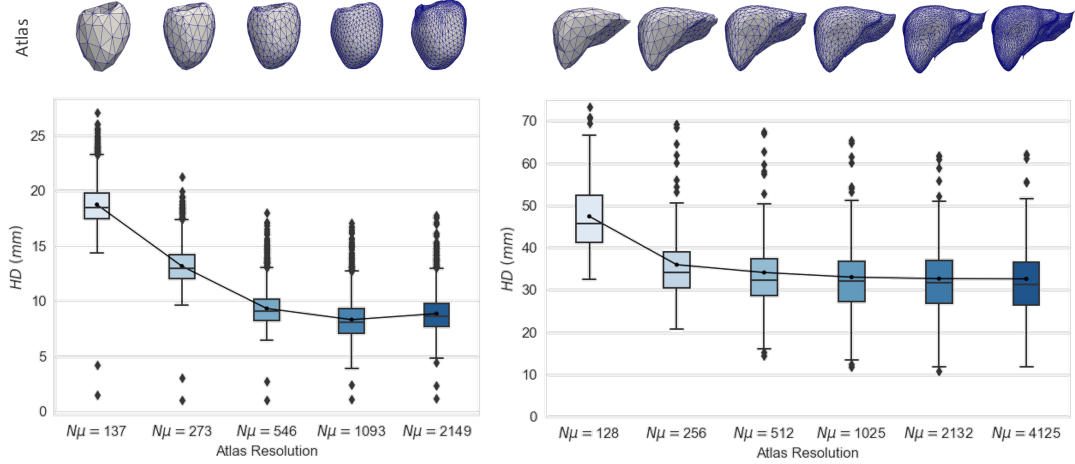


Figure 4.11: Impact of atlas resolution on graph regression on the atlas, and normalising their structures using established vertex-to-vertex correspondences for all population samples of LV and liver datasets. Boxplots with line connecting mean values for each group show the higher resolution atlases outperform the low-resolution ones.

4.3.2 Evaluation of Generation

Given a set of structurally normalised meshes, featuring the same connectivity of the template mesh, a VAE is trained for shape generation. In this section, the performance of the proposed generation model is evaluated by measuring the fidelity of synthetic data in terms of variability and plausibility. By doing so, this section evaluates whether the different generative shape models investigated in this chapter can be used to synthesise virtual cohorts, capturing the variability and plausibility in anatomical shape (i.e. measuring generalisation and specificity metrics), and clinically relevant anatomical indices observed in an actual population. The baseline statistical shape model, named RSMP, which is a registration-PCA-based generative model presented in Chapter 3, is compared with the proposed generative model.

In [93], a comprehensive quantitative analysis was conducted to evaluate the advantages and disadvantages of applying different sampling techniques for PCA-based shape models. This analysis was performed to generate virtual cohorts of the aortic root and vessel. Romero *et al.* concluded that uniform sampling within the learned latent space or principal sub-space increases shape variability within cohorts. However, the cohorts with large variability may contain unrealistic shapes, unrepresentative of native

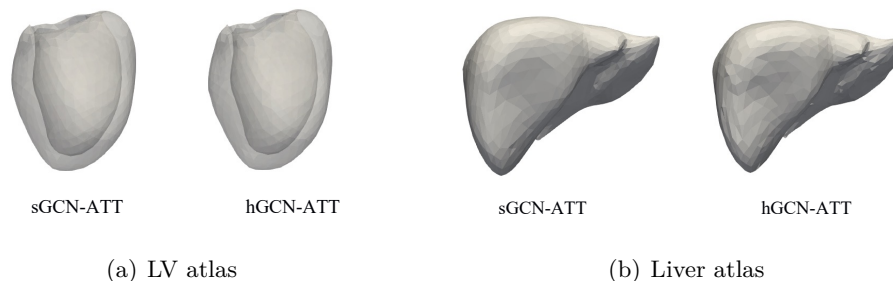


Figure 4.12: The population-derived atlas reconstructed from the (a) LV and (b) liver datasets in two different settings s/hGCN-ATT.

anatomy/naturally occurring variations, as shown in Chapter 3. On the other hand, Gaussian sampling to generate latent vectors that represent virtual shape instances in PCA-based shape models ensures greater plausibility in the synthesised virtual cohorts. Hence, anatomical shape virtual cohorts are synthesised by Gaussian sampling in VAE structure using the generative shape models proposed in this Chapter (i.e. ASMG: (s/h)GCN-ATT-VAE). Similarly, for a fair comparison, virtual cohorts are synthesised by RSMP generative model using Gaussian sampling in PCA.

Generalisability and Specificity

A quantitative assessment in terms of generalisation and specificity is first provided to numerically evaluate the model. Three distance measures Hausdorff distance (HD), minimum Euclidean distance (ED) and its symmetric distance (ED^*) described in Chapter 3, are considered to report generalisation and specificity.

First, the generalisation and specificity metrics for model selection are explored, and the optimum latent dimensionality in the proposed generative model is determined. Model selection identifies the best GCN-ATT-VAE model when applied to two datasets featuring the highest performance in the synthetic shape generation. More specifically, models that demonstrate concurrently low specificity and generalisation errors are generally desired (models positioned closer to the lower left corner in the generalisation-specificity planes). In Figure 4.13, by varying the dimension of the latent space (L), three distance measures HD , ED and ED^* report the model specificity and generalisation (in the sGCN-ATT-VAE setting). For each model, the markers, dotted lines, and arrows indicate the associated average, confidence ellipse, and the

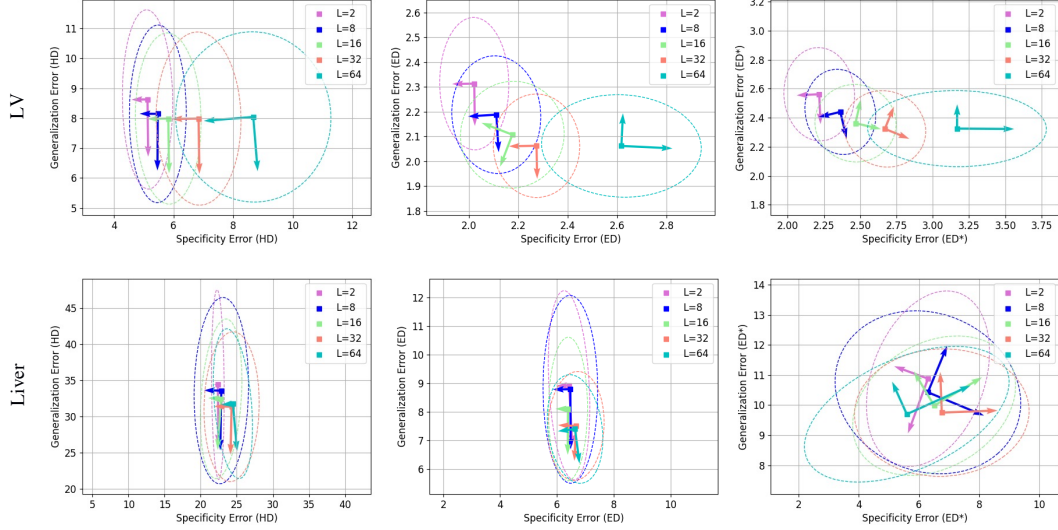


Figure 4.13: Generalisation and Specificity errors (in [mm]) of the ASMG model with different values of latent dimension L . The first and second rows illustrate results on LV and liver datasets, respectively. The plots show the performance of the models with L values smaller than 64. In each row, HD , ED and ED^* distances are reported from left to right. For each model, the markers, dotted lines and, arrows indicate the associated average, confidence ellipse, and the direction of the maximum variability (in error values for test and synthetic data) respectively, for training with various latent dimensions. A high-performance model illustrates simultaneous small generalisation and specificity errors.

direction of the maximum variability in error values respectively. As shown, in the majority of panels, for $L = 16$ a more specific model with high generalisation ability is selected. While there is a tendency for both generalisation and specificity errors to decrease as shown in the direction of error variability.

Furthermore, in Tables 4.5 and 4.6, the generative performance of the ASMG model is compared with the baseline statistical shape model RSMP. A fair evaluation was ensured by retaining the same latent dimension ($L = 16$) for baseline generative models. Table 4.5 summarises the generalisation ability for ASMG methods investigated in this chapter and the RSMP baseline model, for LV and liver datasets. It is observed that for two datasets, the (s/h)GCN-ATT-VAE models significantly outperform RSMP in the majority of distances. This is because our attention-based shape matching maintains

Table 4.5: Generalisation ability: comparison between generative models in terms of generalisation ability using three distance metrics HD , ED and ED^* (mean \pm std) in $[mm]$. **Bold** values present their performance are significantly better than those of the RSMP model, with a p-value < 0.001 .

		ASMG		
		RSMP	sGCN-ATT-VAE	hGCN-ATT-VAE
LV	HD	8.02 ± 2.14	7.99 ± 1.83	7.50 ± 1.50
	ED	2.15 ± 0.20	2.11 ± 0.14	2.28 ± 0.18
	ED^*	2.38 ± 0.24	2.36 ± 0.17	2.50 ± 0.15
Liver	HD	35.94 ± 14.61	32.44 ± 6.86	31.86 ± 6.50
	ED	7.72 ± 2.24	8.09 ± 1.56	7.22 ± 1.8
	ED^*	10.51 ± 3.83	9.98 ± 1.44	8.62 ± 1.91

high accuracy in the training normalised shapes, therefore the generative model presents lower errors when reconstructing unseen actual test data. Also, compared to PCA, which is a linear projection of shapes onto lower-dimensional subspaces, our generative model is based on graph convolution networks and β -VAE, which can capture non-linear variations in shapes. Therefore, (s/h)GCN-ATT-VAE models are capable of capturing the higher variability of the seen (training) data, and generalising to, or explaining, unseen (testing) data.

In Table 4.6, specificity errors for LV and liver datasets are presented in three distance metrics HD , ED and ED^* , in order to assess the anatomical plausibility of synthesised shapes obtained by (s/h)GCN-ATT-VAE and RSMP generative models. It is observed that the highest specificity (i.e. lower specificity errors) of all methods investigated, across both LV and liver structures, is achieved by the ASMG model. Due to lower specificity errors, our similarity-based generative model presents more realistic synthesised shapes when compared with the model based on registration. The lower specificity error can be explained as follows: considering shapes as graphs support our unsupervised shape matching framework to learn better disentangled latent representations and thus derive a more effective form of soft correspondences between shapes, which in turn preserves more details during the normalisation process. As a result, shapes generated from structurally normalised populations have a greater degree of

Table 4.6: Specification ability: comparison between generative models in terms of specificity using three distance metrics HD , ED and ED^* (mean \pm std) in $[mm]$. **Bold** results present their performance is significantly better than those of the RSMP model, with a p-value < 0.001 .

		ASMG		
		RSMP	sGCN-ATT-VAE	hGCN-ATT-VAE
LV	HD	6.57 ± 0.69	5.82 ± 0.70	5.60 ± 0.72
	ED	2.34 ± 0.20	2.18 ± 0.13	2.19 ± 0.15
	ED^*	2.62 ± 0.19	2.47 ± 0.18	2.50 ± 0.12
Liver	HD	31.84 ± 3.53	24.83 ± 1.80	24.51 ± 1.73
	ED	8.53 ± 0.79	6.59 ± 0.43	6.50 ± 0.34
	ED^*	7.09 ± 2.97	6.50 ± 1.68	8.28 ± 0.58

plausibility.

Figure 4.14 visualises the general model performance of three models for the same values of L on the LV (top row) and the liver datasets (bottom). Distances ED/ED^* measure how two shapes are similar on the average basis but are not suitable to detect differences in the details of g_k and g'_k , while Hausdorff distance is a lightweight yet nuanced similarity measure to assess the closeness of two graphs. Therefore, when it comes to drawing conclusions about the general model performance, the Hausdorff distance is a more appropriate metric. The results show that in the majority of the average distances, the (s/h)GCN-ATT-VAE model (especially with hybrid representation) outperforms the statistical shape model RSMP, and obtained the highest performance, due to smaller concurrent specificity and generalisation errors.

Moreover, the average distances (i.e. square markers) in specificity metric are significantly lower for all trained (s/h)GCN-ATT-VAE models compared to others, (p-value < 0.01), indicating that the accuracy of newly generated instances by generative (s/h)GCN-ATT-VAE model are higher and this model presents more realistic synthetic shapes in both datasets.

Purple ellipses are wider than other ellipses in most panels, indicating greater variability in specificity errors for the generated samples using RSMP models. The RSMP generative model is therefore less robust when it comes to synthesising shapes.

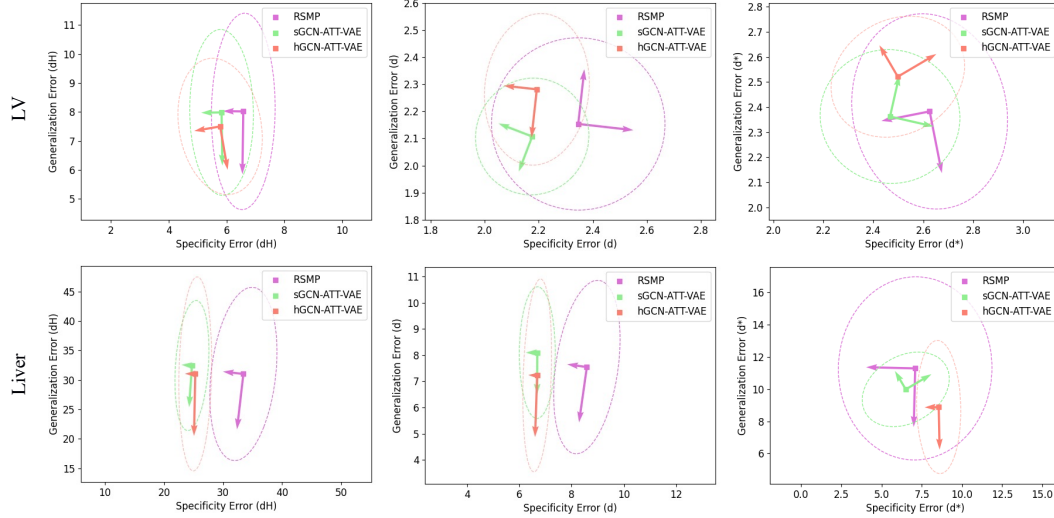


Figure 4.14: Generalisation and Specificity errors (in [mm]) of the RSMP and s/hGCN-ATT-VAE models with same values of latent dimension $L = 16$. The first and second rows present the results on LV and liver datasets, respectively. In each row, HD , ED and ED^* distances are reported from left to right. For each model, the markers and dotted lines indicate the associated average and confidence ellipse, and the direction of the maximum variability (in error values for test and synthetic data) respectively. A high-performance model illustrates simultaneous small generalisation and specificity errors.

As indicated by its high specificity error (average value) in the majority of panels, the RSMP model generates less realistic samples for LV and liver shapes than the other generative models. This is because the CPD computes a transformation while encouraging the displacement vectors to point in similar directions. Thus, to find a dense correspondence between g_1 and g_2 , matching be found with only a subsample of g_1 and g_2 and can be applied on all points. In contrast, s/hGCN-ATT computes the attention-derived correspondences using learned nodal embedding vectors from a deep spatial graph convolution network; which not only aggregates local features through the convolution filter operators in the spatial domain but also captures the global structural information by using deep architecture. Based on the findings, it can be inferred that even with a smaller training set (such as the liver dataset), the (s/h)GCN-ATT-VAE method can produce acceptable generation results. This is attributed to the generative

VAE model’s ability to capture intricate global shape variations, in contrast to the statistical shape models relying on PCA-based approaches.

Of particular importance to note is that hGCN-ATT-VAE, which estimated correspondences based on hybrid representations of shape also shows lower generalisation errors than sGCN-ATT-VAE model, which utilised only spatial features for shape matching. This indicates that the hGCN-ATT-VAE models can capture a greater degree of shape variability for each dataset than afforded by RSMP, and correspondingly can synthesise more diverse (in terms of shape) virtual shape populations than the latter.

Clinical Relevance

After performing the generative models on the actual cohort G of size K , a synthetic cohort G^* with the same size has been generated. For all the anatomical shapes in the synthetic cohorts, the biomarkers (i.e. volume indices) are computed.

Given the actual cohort $G = \{g_k\}_{k=1}^K$, the different acceptance rates \mathcal{A} (described in Section 3.2.2) are computed to measure the percentage of synthetic samples in virtual cohorts G^* that contain volume indices, within a confidence interval of the volume indices distribution observed in the real population. In addition, the measurement of similarity between shapes refers to the evaluation of likeness between the volumes. The resulting volume indices distributions are presented in Figure 4.15 by means of violin charts, for LV and liver datasets. In this figure, the distributions of biomarkers are shown on the actual samples, alongside with those generated using the different methods: RSMP, sGCN-ATT-VAE and hGCN-ATT-VAE. As it is mentioned earlier in this chapter, the statistical shape model RSMP used the Gaussian sampling strategy.

As shown in this figure, horizontal lines indicate the boundaries of the different acceptance criteria: \mathcal{A}^r , with dotted lines, \mathcal{A}^μ , with dash-dotted lines, and \mathcal{A}^M with dashed lines. Comparing actual to synthetic distributions for volume variables show that our ASMG generative model (i.e. s/hGSN-ATT-VAE) generates more clinically relevant and realistic samples from the population of the LV and liver while capturing sufficient variability.

For the liver dataset, the method shows reliable results but can not generate a large variety of samples. It is because the liver dataset includes a wide range of volumetric variation in a small training set which imposes more challenges in generative model training.

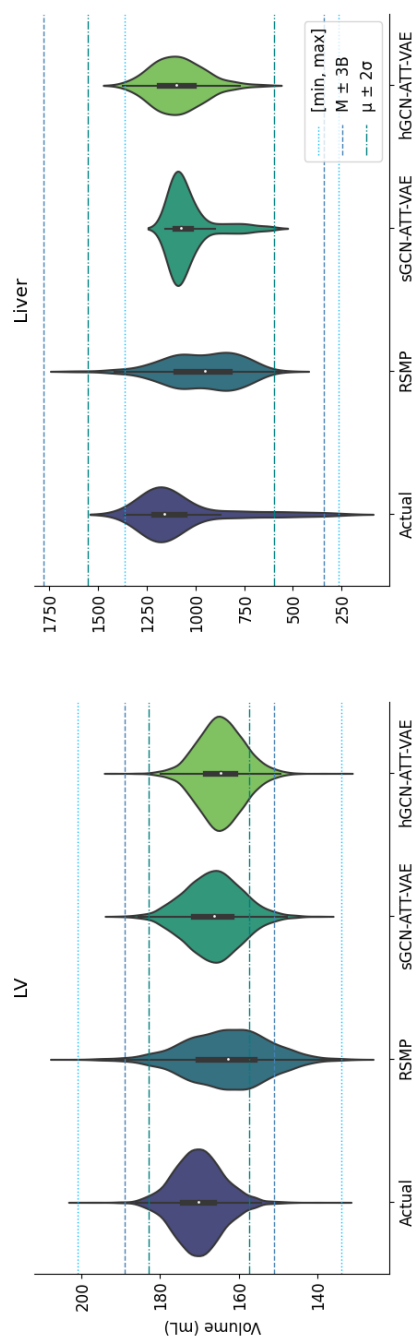


Figure 4.15: Violin plots for the distribution of LV and liver volume indices on the actual samples, alongside with those generated using different methods: RSMP, sGCN-ATT-VAE, and hGCN-ATT-VAE. Horizontal lines mark the confidence intervals for the different acceptance criteria.

Table 4.7: Clinical acceptance rates \mathcal{A} [in %] achieved by different generative models for LV and liver volumes.

		ASMG		
		RSMP	sGCN-ATT-VAE	hGCN-ATT-VAE
LV	$\mathcal{A}^r : \mathcal{A}_{[\min, \max]}$	99.75	100	99.98
	$\mathcal{A}^M : \mathcal{A}_{M \pm 3B}$	86.75	98.10	98.67
	$\mathcal{A}^\mu : \mathcal{A}_{\mu \pm 2\sigma}$	65.85	88.65	88.85
Liver	$\mathcal{A}^r : \mathcal{A}_{[\min, \max]}$	96.40	100	100
	$\mathcal{A}^M : \mathcal{A}_{M \pm 3B}$	100	100	100
	$\mathcal{A}^\mu : \mathcal{A}_{\mu \pm 2\sigma}$	98.56	100	100

Also, despite further morphological variability across the liver shapes, our model considers a unique atlas shape, a limitation that will be addressed in Chapter 5. PCA-based generative model RSMP, with higher generalisation and specificity errors, is reflected in the unrealistic distributions observed for volume indices in the synthesised virtual populations. The figure shows that the statistical shape model RSMP generates distributions of the LV and liver volumes that surpass the range defined by \mathcal{A}^r , higher variability than that observed in the actual cohort.

Table 4.7 shows the efficiency of each method measured using the different acceptance criteria. The clinical acceptance rates of LV and liver virtual cohorts synthesised by our generative model are higher than the statistical shape RSMP method. The higher specificity of the ASMG generator models is complemented by their ability to better preserve key clinical indices in the synthesised virtual cohorts than the RSMP model, as evidenced by the higher clinical acceptance rates achieved for each relevant index evaluated (refer to Table 4.7). Synthetic liver shapes generated by the ASMG model meet all defined criteria, making them clinically relevant. The results demonstrate that the acceptance rates of hGCN-ATT-VAE model across the LV volume indices are consistently higher than those of sGCN-ATT-VAE. This confirms the results in Figure 4.14 that our generative model when considering "spatial" and "surface" information (i.e hGCN-ATT-VAE) outperforms those that only rely on spatial features (i.e sGCN-ATT-VAE). Hence, spatial position alone is inadequate as a descriptor for shape matching and generation. The hGCN-ATT-VAE framework is therefore more suitable

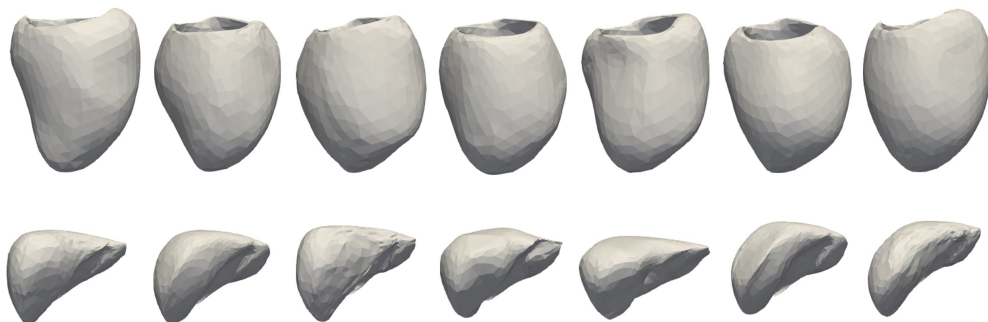


Figure 4.16: Examples of virtual (LV/liver) samples generated by the ASMG generator model.

for ISCTs that require higher statistical fidelity based on anatomical characteristics.

Figure 4.16 provides a visualization of the generated (synthetic) samples by the ASMG model, showcasing anatomical shapes that are realistic at an individual level.

4.4 Conclusion and Limitations

In this chapter, a novel unsupervised probabilistic deep generative model is proposed for generating virtual anatomical shapes from training shapes of various topologies. The lack of consistent tensor-like representation across the training shapes poses significant methodological challenges that were aimed to be addressed here. The structures of the shapes within the context of the graphs were considered, and an unsupervised graph-based generative model was developed to generate numerous realistic synthetic shapes from graph datasets with no vertex-to-vertex correspondences.

In contrast to existing works on shape generative modelling that requires datasets with fixed topology and supervised vertex correspondences, an unsupervised deep learning framework was proposed to establish dense shape correspondences across the training data in the latent space. This approach implicitly solves the shape correspondence problem in the latent space and eliminates the need for initial shape correspondence. This happened by considering the connectivities between vertices via latent vertex representations derived from a spatial graph convolutional network. The latent vertex representations are used to iteratively compute the attentions between the vertices of the training samples and a template mesh, and subsequently regress the input meshes onto the latter. The convergence of this procedure results in a set of structurally nor-

malised meshes, featuring the same connectivity of the template mesh, which are used to train a variational autoencoder for data generation.

The study demonstrated the suitability and applicability of the proposed generative model in conducting ISCTs, with virtual cohorts, through a comparative analysis. The results indicated that hybrid representations of shapes provided substantial improvements over spatial representation in terms of anatomical validity in the estimated correspondences. This improvement can be attributed to the network’s ability to explicitly encode both spatial and surface information of the shapes. It was shown that considering vertex connectivities (i.e., representing shapes as graphs rather than point sets) leads to improved shape correspondences and generation compared to point-based registration methods. By modelling meshes as graphs, the proposed framework exploited a richer representation due to the edge information and addressed limitations observed in the traditional point set based methods. Therefore the proposed attention-based shape matching outperformed registration-based methods.

The model’s versatility was studied by applying it to a dataset with a wide range of variability across the provided population, and the influence of data scarcity on model performance was explored. The framework was validated on real image-derived clinical datasets that included a large range of LV cardiac data and a small dataset of liver samples. The results summarized the ability of the proposed framework to generate realistic LV meshes, based on a large training population from UKB. Importantly, it also showed reliable results for liver shapes, given a much smaller training population, and in the presence of relatively large morphological variations. These results, moreover, were significantly better than those based on PCA. This is because the VAE is a non-linear probabilistic generator and is able to capture the shape variations with a higher flexibility.

In this proposed framework, a refinement strategy on node embeddings was not considered. These features are obtained in the deep GCN and thus can be task-optimised in a more general end-to-end framework, to preserve more morphological details in normalised shapes and increase the plausibility of virtual samples. Although the proposed framework enables learning from graph populations featuring disparate connectivities, the generated meshes have the same connectivity, defined by the atlas. Our method considered a unique atlas for each shape category and regressed all shapes, regardless of morphological variations on the atlas. This certainly will have increased distance

4.4 Conclusion and Limitations

errors and diminished fidelity. These limitations will be addressed in Chapter [5](#).

CHAPTER 5

An End-to-End Deep Learning Framework for
Refinable Shape Matching and Generation:
Enhancing Specificity and Generalisability

5.1 Introduction

The proposed ASMG shape generative framework in Chapter 4 constructed a template/atlas in the form of a mean shape (or mesh) and structurally normalised different-sized observed shapes with the option of leveraging a registration of the observed shape to the atlas. However, the framework can be improved by considering some points, which will result in higher performance.

The matching process in ASMG does not consider a refinement strategy for embeddings to establish correspondences between nodes. Aside from that, in the regression process (for graph normalisation), the distance between the constructed atlas and each observed shape is taken into account regardless of how closely the normalised shape resembles the actual shape. Consequently, some morphological details may be overlooked during graph normalization, affecting the variability and plausibility of synthesised shapes. These limitations are addressed in this chapter.

The Atlas Refinable Attention-based Shape Matching and Generation network (AtlasR-ASMG) is proposed here. It is a framework that allows for the joint learning of high-quality refinable shape matching and generation on 3D surface mesh data while also constructing a population-derived atlas model during the process.

The contributions include:

- An all-in-one deep learning framework for atlas construction, shape matching and generation using geometric deep learning and attention mechanism;
- A robust refinable attention-based shape matching system which improves over baseline shape matching models, relying on both geometric and surface features of the shape;
- A method for on-the-fly construction of a population-derived atlas within a deep learning framework;
- An unsupervised probabilistic deep learning framework to compute vertex-wised correspondences among the shapes, and generate high quality meshes in the atlas domain, without significantly sacrificing shape details.
- Generating plausible virtual population of shapes from different-sized shapes.

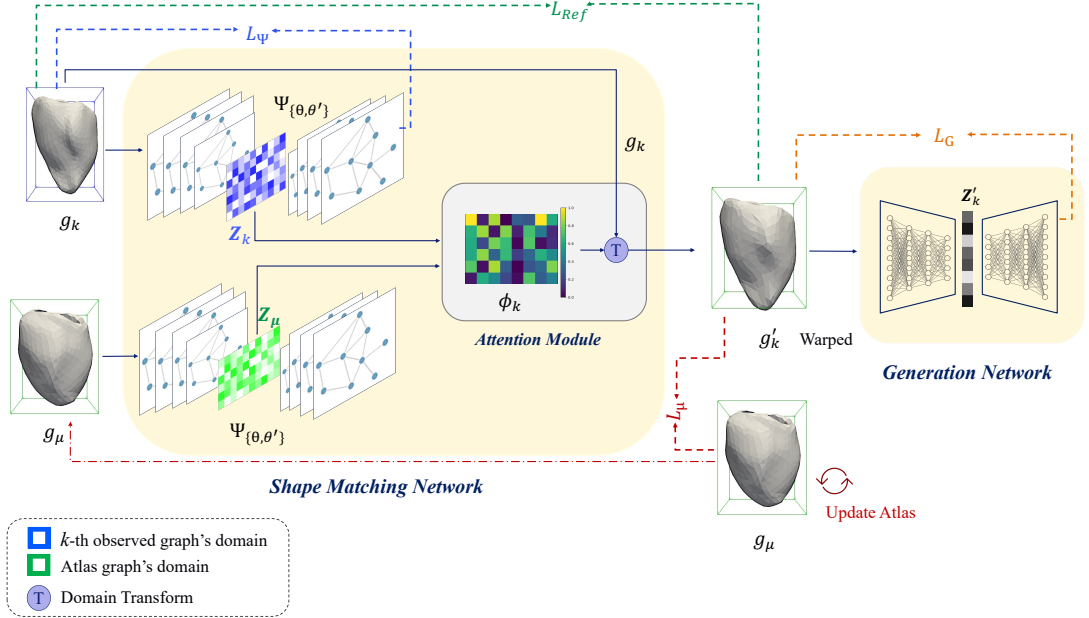


Figure 5.1: Overview of the Atlas Refinable Attention-based Shape Matching and Generation network (Atlas-R-ASMG) framework for shape generation from different-sized shapes.

- The model possesses the capability to function as a joint clustering generative model, implying that it combines both clustering and generative modelling techniques within a unified framework. By extending as a joint clustering generative model, the model gains the ability to leverage the complementary strengths of clustering and generative modelling, leading to improved performance and more comprehensive analysis in various applications.

Although the proposed generative framework is demonstrated here on the left ventricle and liver structures, it is generic in design and can be applied similarly to other anatomical shape ensembles.

5.2 Atlas-R-ASMG model

Problem Statement: A graph $g = (V, \mathbf{A}, \mathbf{X})$ consists of a finite set of nodes $V = \{1, 2, \dots, N\}$, an adjacency matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$. Suppose each vertex has a D -dimensional feature vector, we use $\mathbf{X} \in \mathbb{R}^{N \times D}$ to denote the graph's node feature

matrix with each row representing a vertex.

Consider a graph dataset $G = \{g_k\}_{k=1}^K$ containing K shapes with different cardinalities. Each g_k represents a 3D surface shape with cardinality $|V_k| = N_k$. The goal is to train a learning framework that promotes two critical characteristics of the generated shapes: *high-quality shape matching* and *anatomical plausibility*. The former improves the quality of the correspondence map through the nodal embedding while the latter ensures that each normalised shape in the atlas domain is as close to its actual counterpart as possible, and thus promotes the plausibility of randomly synthesised shapes.

This framework is presented for endowing deep generative models with anatomical plausibility reasoning. Atlas-R-ASMG extends the ASMG framework (in Chapter 4) to jointly construct a population-derived atlas while training a model to perform high-quality attention-based shape matching and generation using a refinement strategy. As illustrated in Figure 5.1, the Atlas-R-ASMG pipeline comprises two primary components, *shape matching* and *generation*, which are jointly trained in an end-to-end manner. The initial component of the pipeline involves establishing correspondence maps across the training data using an unsupervised probabilistic refinable attention-based shape matching network. This process simultaneously finds the atlas shape and warps it into each observed shape, referred to as normalisation of shape structures (i.e., domain transformation) within the 3D space. By doing so, the atlas topology is preserved. This normalisation procedure ensures a consistent representation of shapes across different instances, accommodating structural variations, and facilitating effective shape matching. The second component involves a generative network that learns a probability density function based on a set of structurally normalised shapes. This generative network captures the underlying distribution of the normalised shapes in the 3D space, enabling the generation of new shapes that adhere to the learned distribution.

To ensure topological plausibility and accuracy, a refinement procedure is incorporated into the pipeline (see the green dotted line). This iterative refinement step addresses errors in vertex-wise correspondences and shape normalisations, progressively improving the alignment and matching of shapes. The refinement strategy plays a crucial role in enhancing the overall performance and accuracy of the pipeline, particularly in scenarios where a single pass of the model without refinement may not be sufficient.

Through the end-to-end training process, the Atlas-R-ASMG pipeline effectively learns to fit a population-derived atlas shape to the training data. This population-derived atlas serves as a comprehensive reference template, encapsulating the common features and variations observed within the dataset. The integration of the refinement procedure and the population-derived atlas contributes to improved shape matching, generation, and overall performance, surpassing the capabilities of a single pass of the Atlas-R-ASMG model.

5.2.1 Refinable Attention-based Shape Matching (R-ASM)

Shape matching refers to the problem of establishing meaningful structural correspondences of nodes between two or more shapes by taking both node similarities into account. The shape matching procedure is modeled in close analogy to related approaches in Chapter 4, by computing *refinable* similarities between nodal embeddings in the atlas shape and the observed shape.

Feature extraction network

Analogous to previous approaches in Chapter 4, graph convolutional neural (GCN) network Ψ performs feature extraction. The purpose of this network is to learn embedding features from an input shape which is useful for the normalisation task (normalising shapes structures by warping them to an atlas shape). The feature extraction network is built by exploiting the ability of spatial-based geometric deep learning methods to handle inconsistent shape populations, where the convolution is performed in local Euclidean neighborhoods.

Hence, analogous to Section 4.2.1.3 variational graph auto-encoder models with spatial-based graph convolution layers are used to build the network Ψ (parametrised by trainable weights $\{\theta, \theta'\}$). It takes the adjacency matrix \mathbf{A} and node features \mathbf{X} as input and tries to recover the feature matrix \mathbf{X} through the hidden layer embeddings \mathbf{Z} . Given the template/atlas g_μ and observed shapes g_k , nodal embeddings \mathbf{Z}_μ and \mathbf{Z}_k are computed by a shared graph neural network Ψ as

$$\{\mathbf{Z}_k, \mathbf{Z}_\mu\} = \Psi_{\{\theta, \theta'\}}(g_k, g_\mu) \quad (5.1)$$

where

$$\mathbf{Z}_k = \Psi_\theta(\mathbf{X}_k, \mathbf{A}_k) \quad \text{and} \quad \mathbf{Z}_\mu = \Psi_\theta(\mathbf{X}_\mu, \mathbf{A}_\mu). \quad (5.2)$$

This inference model calculates embeddings via the spatial-based graph convolutional layers and each layer performs graph convolution locally on each node by Equation 2.21. That is, the convolution operator learns features from the preceding network layer to dynamically determine the association between filter weights and graph neighbourhood, rather than relying on static predefined local pseudo-coordinate systems.

Similar to Chapter 4, the feature extraction network Ψ can be presented in two different settings: sGCN and hGCN. The former refers to where representations of vertices are only defined by spatial features, while the latter serves as hybrid representations of vertices defined by spatial positions of points, with the associated vertex normal vectors.

Attention module

Once the graph embedding is learned, an attention mechanism is applied to those embedded features to learn correspondence across the population. Domain transformation can therefore be performed using the established correspondences. Thus, an ATT module is proposed to learn the mapping from the node function space $\mathcal{F}(\mathbb{R}^{N_k})$ to $\mathcal{F}(\mathbb{R}^{N_\mu})$ and to project shapes to the atlas domain as,

$$g'_k = ATT(\underbrace{\mathbf{Z}_k, \mathbf{Z}_\mu}_{\phi_k}, g_k), \quad (5.3)$$

where $g_k \in \mathcal{F}(\mathbb{R}^{N_k})$ and $g'_k \in \mathcal{F}(\mathbb{R}^{N_\mu})$. Each structurally normalised shape g'_k is presented by matrix feature representation \mathbf{h}'_k . More specifically, the soft correspondence (i.e. attention maps) $\phi_k \in [0, 1]^{N_\mu \times N_k}$ is obtained as the mapping function in the embedded-space paradigm by $\Phi_k = \text{Softmax}(\lambda \mathbf{Z}_\mu \mathbf{Z}_k^T)$ (Equation 4.2), given latent node embeddings $\mathbf{Z}_k \in \mathbb{R}^{N_k \times F}$ and $\mathbf{Z}_\mu \in \mathbb{R}^{N_\mu \times F}$. Subsequently, this module uses a soft attention mechanism as a shape domain transformer, which normalises shape structures using Equation 4.3. Therefore all the samples of training shapes will structurally be normalised into the function space $\mathcal{F}(\mathbb{R}^{N_\mu})$.

In the training phase, a refinement strategy is employed to avoid finding false correspondences and present plausible shapes in the atlas domain. Indeed, the refinement procedure is used to optimise the GCN network (Ψ) to improve the quality of correspondences and the fit of the normalised shapes in relation to the actual shapes. The details are described in Section 5.2.3.

5.2.2 Generation Network

Next, a variational auto-encoder as a generation network G (parameterised by $\{\psi, \psi'\}$) is trained using the structure normalised shapes to generate synthetic shapes g^{syn} as;

$$\{g^{syn}\} = G_{\{\psi, \psi'\}}(g'_k). \quad (5.4)$$

This network consists of fully-connected layers in a β -VAE format. Novel shape samples are generated by simply sampling from the prior normal distribution and using the sample as input to the decoder $G_{\{\psi'\}}$, which converts it into a random synthetic shape with the same connectivity as the atlas.

A forward pass through the network via feature extractor network (Equation 5.1), Attention module (Equations 5.3) and Generation network (Equation 5.4) can be expressed concisely as:

$$g^{syn} = G_{\{\psi, \psi'\}} \left(ATT \left(\Psi_{\{\theta, \theta'\}}(g_k, g_\mu), g_k \right) \right) \quad (5.5)$$

A graphical representation of the Atlas-R-ASMG generative model is shown in Figure 5.15(a), showing the hypothesised dependencies between variables.

5.2.3 Unsupervised Loss

The overall unsupervised loss is composed of three parts: feature extraction loss (L_Ψ), refinement loss (L_{Ref}), and generation loss (L_G);

$$L = L_\Psi + L_{Ref} + L_G \quad (5.6)$$

where L_Ψ and L_{Ref} are associated with shape matching procedure and L_G refers to generation process (see Figure 5.1). In the following, each loss term is explained in more detail. Refinement refers to the optimisation of the shape matching network weights in the training phase for establishing correspondences and normalising the shape with arbitrary cardinality (e.g. atlas's cardinality).

Feature extraction loss: An ELBO loss is used for unsupervised learning of the embedding network Ψ , i.e.,

$$L_\Psi(\theta, \theta') = \begin{cases} \frac{1}{2} \sum_{k=1}^K \sum_{i=1}^{N_k} \|\mathbf{x}_{ki}^{rec} - \mathbf{x}_{ki}\|^2 - w_{kl} D_{KL}[q\theta(\mathbf{Z}_k | \mathbf{X}_k, \mathbf{A}_k) \| p(\mathbf{Z}_k)] & \Psi:\text{sGCN} \\ \frac{1}{2} \sum_{k=1}^K \sum_{i=1}^{N_k} \|\mathbf{x}_{ki}^{rec} - \mathbf{x}_{ki}\|^2 - w_{kl} D_{KL}[q\theta(\mathbf{Z}_k | \mathbf{X}_k, \mathbf{A}_k) \| p(\mathbf{Z}_k)] \\ \quad + w_{norm} \sum_{p \in g_k^{rec}} (1 - \|\mathbf{n}_{kp}\|^2) & \Psi:\text{hGCN} \end{cases} \quad (5.7)$$

where network Ψ (named sGCN) learns embeddings of the spatial information and node feature matrix \mathbf{X} denotes the Cartesian geometry of nodes, whose i -th row is $\mathbf{x}_i = (x_i, y_i, z_i)$. The network Ψ (named hGCN) learns combined embeddings of geometry and surface. The feature matrix \mathbf{X} is a horizontal concatenation of vertex geometries and vertex normals and i -th row represented by $\mathbf{x}_i = (x_i, y_i, z_i, \mathbf{n}_i^x, \mathbf{n}_i^y, \mathbf{n}_i^z)$ where \mathbf{n} denotes a normal vector. Hyperparameters w_{kl} and w_{norm} are set empirically.

Optimising the network Ψ , the model can provide the normalised shapes g'_k . However, unlike the ASMG model in Chapter 4, Atlas-R-ASMG model minimises an additional loss (L_{Ref}) to refine these structurally normalised shapes.

Refinement loss: Shape matching incorporates a refinement strategy to avoid finding false correspondences while projecting shapes onto the atlas domain. To achieve this, the loss L_{Ref} encourages accurate transformation of shape g_k to the atlas domain by:

$$L_{Ref} = \sum_{k=1}^K w_{cd} CD(g_k, g'_k) + w_{lap} L_{lap}(g'_k) \quad (5.8)$$

where Chamfer Distance (CD) [122] measures the distance of vertices between two graphs: $CD(g, g') = \sum_{\mathbf{x}_i \in g} \min_{g'} \|\mathbf{x}_i - \mathbf{h}'_j\|^2 + \sum_{\mathbf{h}'_j \in g'} \min_g \|\mathbf{x}_i - \mathbf{h}'_j\|^2$. The Laplacian loss L_{lap} is a regularization term that encourages neighbouring vertices to move coherently, reducing mesh self-intersections and allowing for smoother surface reconstructions [123]. Given j -th vertex on k -th graph, and neighbouring vertices q , where $q \in \mathcal{N}_{kj}$, it is defined as:

$$L_{lap} = \sum_{k=1}^K \sum_{j=1}^{N_\mu} (\mathbf{h}'_{kj} - \sum_{q \in \mathcal{N}_{kj}} \frac{1}{|\mathcal{N}_{kj}|} \mathbf{h}'_{kq}). \quad (5.9)$$

This is constructed as the difference of the position vertex and the mean over all neighbouring vertices. The weights associated with the terms in the loss function (w_{cd} and w_{lap}) are hyperparameters that are set empirically.

Therefore, the hyperparameters of the shape matching model are adjusted to minimise both losses L_Ψ and L_{Ref} .

Generation loss: In order to generate synthetic shapes the loss function L_G follows the original loss of the β -VAE, where hyperparameter β makes a balance between low reconstruction error and high latent space quality, which emphasizes discovering disentangled latent factors. This generative network derives a pdf from the set of normalised graphs. The probability of node variations is approximated in the latent space $\mathbf{z}'_k \in \mathbb{R}^L$

via a posterior $q_\psi(\mathbf{Z}'_k|\mathbf{h}'_k)$. By drawing samples from this approximate posterior probability, the likelihood of the observed population is estimated. The variational lower bound L_G is optimised w.r.t the variational parameters ψ, ψ' :

$$L_G(\psi, \psi') = \sum_{k=1}^K (\mathbb{E}_{q_\psi(\mathbf{z}'_k|\mathbf{h}'_k)}[\log p_{\psi'}(\mathbf{h}'_k|\mathbf{Z}'_k)] - \beta D_{KL}[q_\psi(\mathbf{Z}'_k|\mathbf{h}'_k) \parallel p(\mathbf{Z}'_k)]), \quad (5.10)$$

where $\mathbf{h}'_k \in \mathbb{R}^{N_\mu \times 3}$ presents graph geometric features after normalisation process (in Section 4.2.1) and the first term is a reconstruction error which computes the squared Euclidean distance (L2-norm) between input and reconstructed shapes by the decoder. D_{KL} denotes the Kullback-Leibler divergence and computes the divergence between the Gaussian prior $N(0, I)$ and posterior distributions of the latent space \mathbf{Z}' and the KL divergence is weighted by β .

5.2.4 Atlas Construction

Inspiration is drawn from the atlas construction in Chapter 4, where the projected shapes are averaged to form a common atlas space and create an atlas. The shape matching and the generation are trained jointly, and the normalised shapes g'_k and atlas shape g_μ are updated at the end of each epoch by warping the training shape to atlas space via a forward pass of Atlas-R-ASMG and averaging across the samples. Similar to Chapter 4, the atlas (represented by the feature matrix \mathbf{X}_μ) is reconstructed by minimising the following cost function, which entails a Laplacian term for smoothness

$$L_\mu = \frac{1}{2} \sum_{k=1}^K \sum_{j=1}^{N_\mu} |\mathbf{h}'_{kj} - \mathbf{x}_{\mu j}|^2 + \frac{\gamma}{2} \sum_{j=1}^{N_\mu} \sum_{q=1}^{N_\mu} a_{jq} |\mathbf{x}_{\mu j} - \mathbf{x}_{\mu q}|^2 \quad (5.11)$$

where a_{jq} are elements of the adjacency matrix \mathbf{A}_μ . Starting from an initial canonical atlas shape $\mathbf{X}_\mu^{(0)}$, a new atlas $\mathbf{X}_\mu^{(i+1)}$ is iteratively computed from $\mathbf{X}_\mu^{(i)}$ according to

$$\mathbf{x}_{\mu j}^{(i+1)} \leftarrow \frac{\sum_k \mathbf{h}'_{kj} + \gamma \sum_{q \in \mathcal{N}_{\mu j}} \mathbf{x}_{\mu q}^{(i)}}{K + \gamma \sum_{q \in \mathcal{N}_{\mu j}} a_{jq}}; \quad \forall j \in N_\mu \quad (5.12)$$

where $\gamma = N_\mu / \max(N_{\mu j})$, we build the atlas shape g_μ with a fixed adjacency matrix \mathbf{A}_μ so that the vertex positions are optimised while preserving the topology of the atlas.

At the end of each training epoch, a forward pass through the network (Equation 5.5) is used to warp each training case to the atlas space. The updated atlas shape at

the end of each epoch, ($i > 0$) in Equation 5.12, is utilised during the following epoch for computing the losses L_{Ψ} (Equation 5.7) and L_{Ref} (Equation 5.8).

Unlike the atlas derivation in the ASMG model (Chapter 4), where an atlas of the training shape is learned during model training, in this model, we simultaneously construct atlases of the training shapes and correspondences during model training.

In contrast to the ASMG model presented in Chapter 4, the current model involves the joint construction of atlases for the training shapes and refined correspondence maps during the model training. The refinement procedure focuses on re-ranking false correspondences and representing more plausible shapes in the atlas domain.

5.3 Results and Discussion

A number of experiments were conducted to evaluate the performance of the generative shape models proposed in this chapter and compare them against each other and the model proposed in Chapter 4 and RSMP in Chapter 3. Left ventricle shapes derived from CMR images available in the UK Biobank and liver shapes derived from the CT image dataset are utilized for training and validating the framework. The experimental setup is similar to the ASMG framework. For both datasets, w_{KL} is set to $1e^{-3}$. For the LV dataset, the hyperparameters w_{cd} and w_{lap} are empirically set to 1. For the liver dataset, $w_{cd} = 1$ and $w_{lap} = 1.2$. The optimal value for β was achieved by empirically decreasing the values from $2e^{-3}$ to $2e^{-6}$ for the LV dataset and from $1e^{-3}$ to $2e^{-3}$ for the liver dataset.

The method is evaluated in terms of both its matching (i.e., correspondence establishment) and generation performances, which analyze the performance of individual components of the method.

5.3.1 Evaluation of Shape Matching

This section investigates the qualitative and quantitative performance of our Refinable Attention-based Shape Matching (R-ASM) method. For the experimental evaluation, two different settings, s/hGCN-ATT, are considered. The obtained results in Chapter 4 demonstrated that using surface information for node representation (in the hGCN networks) results in more accurate features derived from the network, which, in turn, provides a high-quality correspondence. The benefits of the refinement strategy are

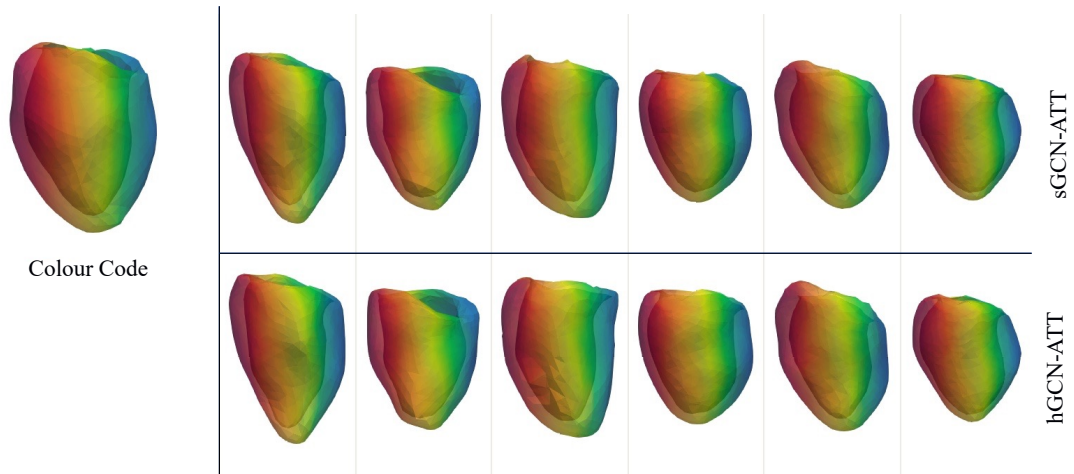


Figure 5.2: Qualitative shape matching results using the R-ASM method on the LV dataset. The atlas shape is shown merely for visualisation purposes (colour code).

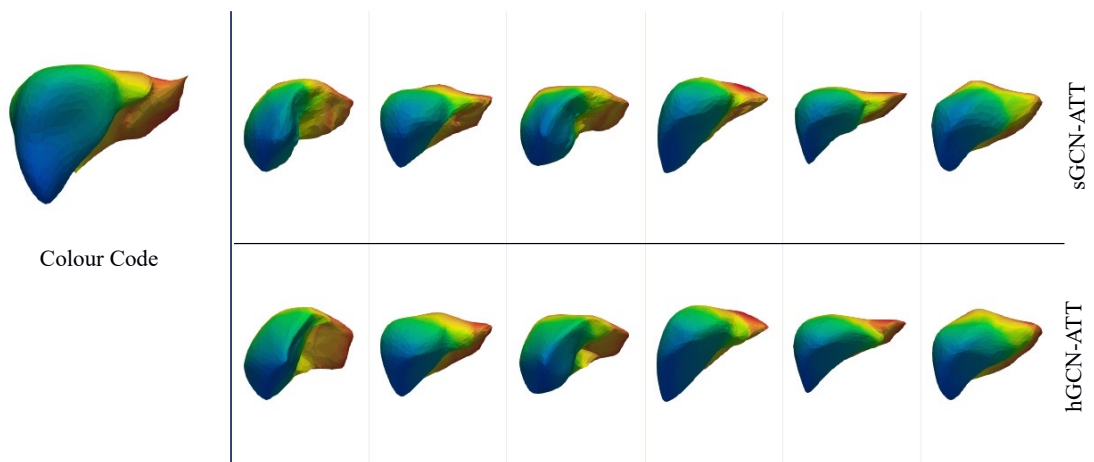


Figure 5.3: Qualitative shape matching results using the R-ASM method on the liver dataset. The atlas shape is shown merely for visualisation purposes (colour code).

highlighted by evaluating the performance of both s/hGCN-ATT models utilising the refinement strategy.

The qualitative results on the LV dataset are shown in Figure 5.2, and the results on the liver dataset are presented in Figure 5.3. The colour-coded shapes in these figures aid in visually understanding the spatial relationships between vertices in different shapes and how they correspond to each other. In these figures, the R-ASM method (in two settings s/hGCN-ATT) predicts vertex-to-vertex correspondences for each shape to obtain shape matchings among a collection of shapes. The corresponding points between the atlas (colour-code) and obtained shapes g' are shown with the same colour.

Additional qualitative results over the five LV and liver cases are summarised in Figure 5.4 and Figure 5.5 respectively. The correspondence maps obtained from different methods are utilised to transform actual input shapes g_k from the domain \mathbb{R}^{N_k} to \mathbb{R}^{N_μ} . Visual inspection of results shows that the improvement was achieved with a refinement strategy over the ASM. By employing a trainable refinement procedure, the R-ASM network effectively captures meaningful similarities between shapes, resulting in more plausible normalised shapes in the atlas domain compared to the obtained shapes by ASM network. This highlights the benefit of deriving refinable nodal embedding and shape matching for both s/hGCN-ATT settings. The R-ASM model in the hGCN-ATT setting achieves the best results, where the normalised shapes g' obtained for cases with either $N_k > N_\mu$ or $N_k < N_\mu$ preserve more anatomical details, avoid distortions, and closely resemble the related actual shape g . The red arrows highlight differences in certain landmarks. In particular, as shown in Figure 5.5, R-ASM in the hGCN-ATT setting effectively captures variation and local details on the more complex dataset (i.e. liver) with substantially fewer training data. It is therefore more suitable where a high degree of fidelity is required based on anatomical characteristics.

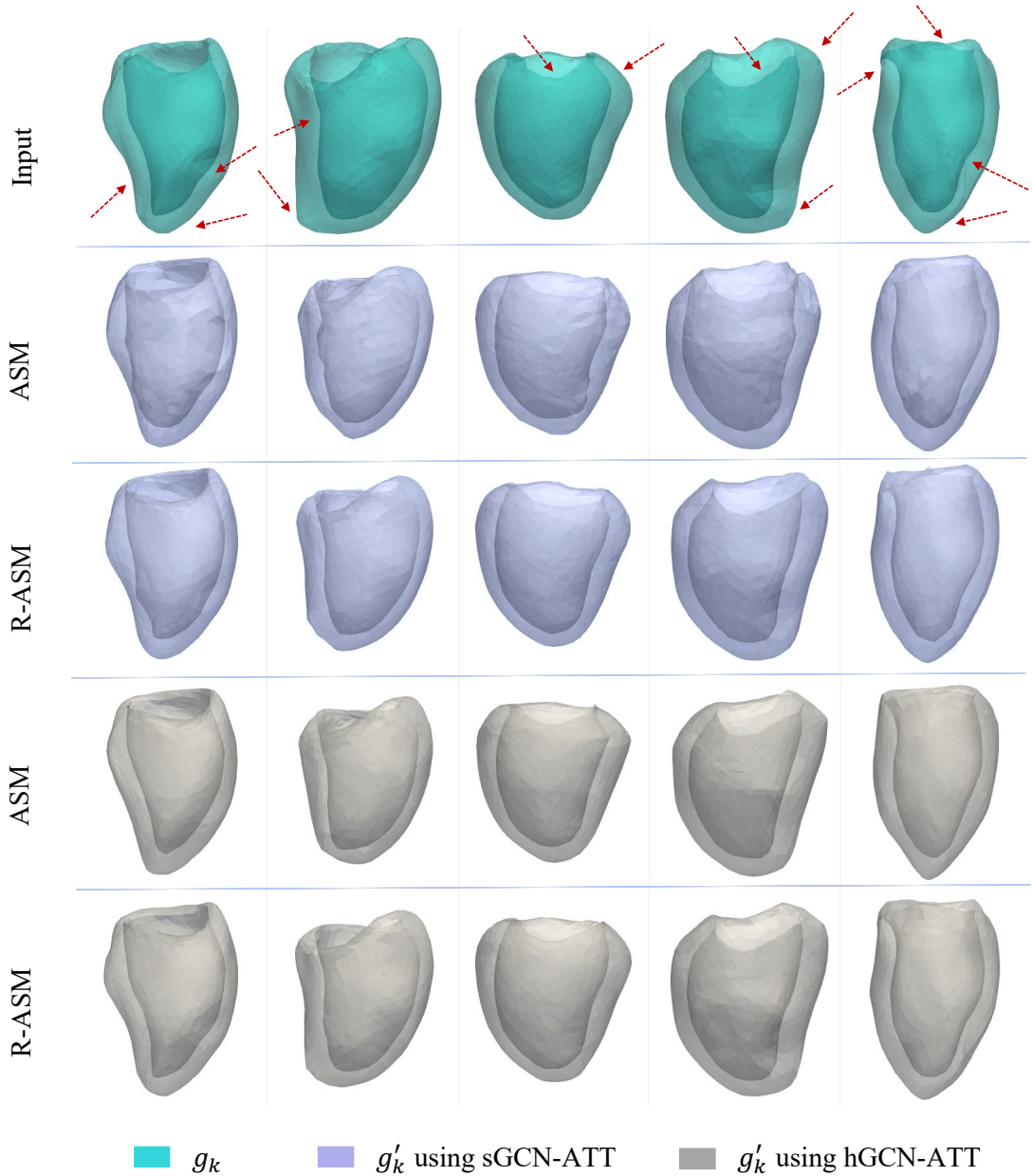


Figure 5.4: Examples show the comparison between different shape matching approaches ASM and R-ASM (in two settings s/hGVCN-ATT) for deriving correspondence on the LV dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1586, 1539, 1455, 1150$ and 1039 respectively from left to right. Shapes g'_k are the normalised meshes with cardinality $N_\mu = 1093$ obtained from different methods in the s/hGVCN-ATT settings. Notice endocardial, epicardial, LV base, and apex landmarks.

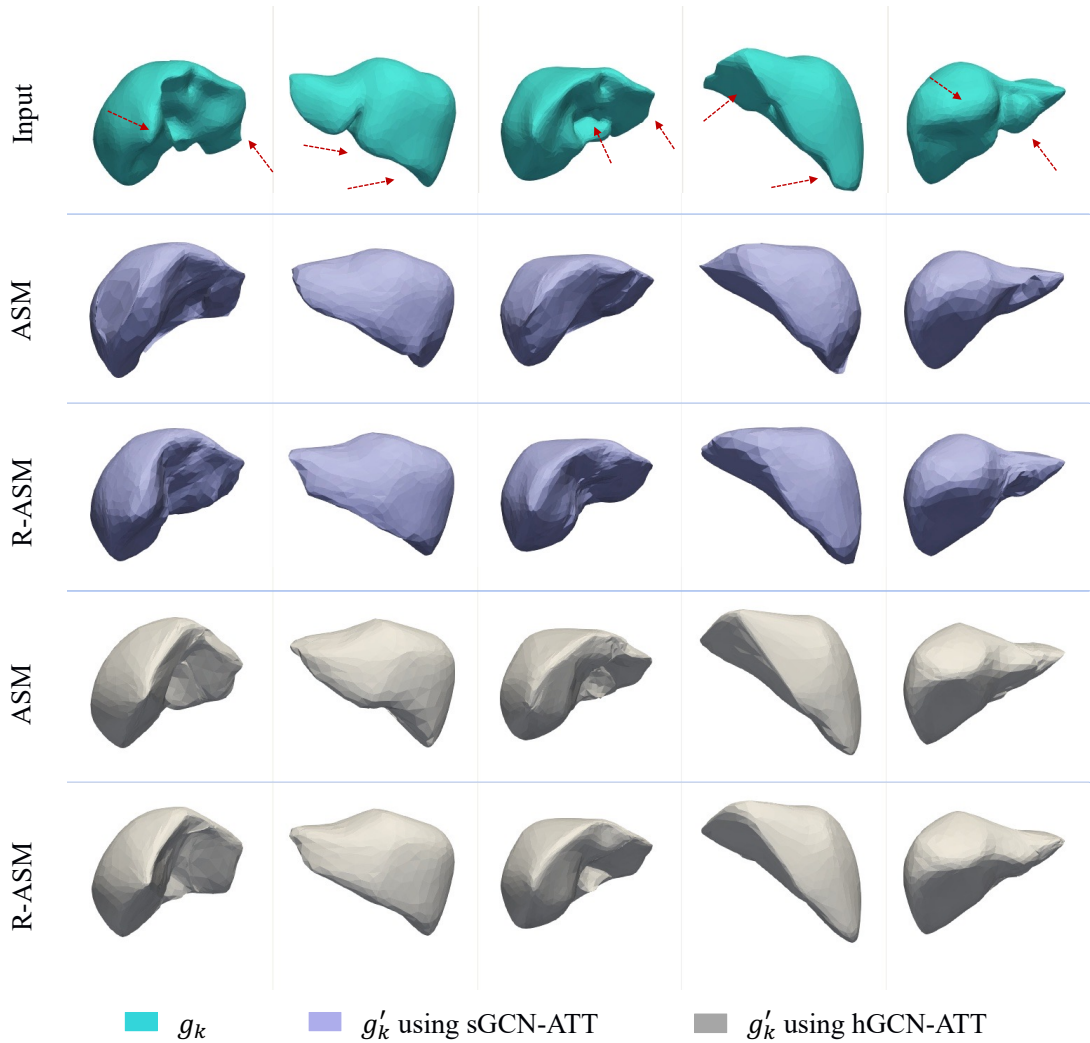


Figure 5.5: Examples show the comparison between different shape matching approaches ASM and R-ASM (in two settings s/hGCN-ATT) for deriving correspondence on the liver dataset. Cyan-coloured meshes present input g_k with cardinality $N_k = 1532, 1338, 1278, 1030$ and 1365 respectively from left to right. Shapes g'_k are the normalised meshes with cardinality $N_\mu = 1025$ obtained from different methods in the s/hGCN-ATT settings. Red arrows highlight some landmarks.



Figure 5.6: The population-derived atlas reconstructed from the (a) LV and (b) liver datasets by Atlas-R-ASM method in two different settings s/hGCN-ATT.

Table 5.1: Shape Matching Quality: comparison between different methods; RSM, ASM and R-ASM using two distance metrics HD and CD (mean \pm std) in $[mm]$. **Bold** numbers are the best and the second best with the best also underlined, for a given metric. Statistically significant (p-value < 0.01) improvement of the best model over a given model is indicated by superscript *.

	ASM		R-ASM	
	sGCN-ATT	hGCN-ATT	sGCN-ATT	hGCN-ATT
LV	HD	8.11 ± 2.13	8.32 ± 1.77	$6.43 \pm 1.44^*$
	CD	12.04 ± 27.63	9.91 ± 1.54	$9.85 \pm 0.94^*$
Liver	HD	35.09 ± 14.55	32.79 ± 10.16	$27.30 \pm 7.71^*$
	CD	254.46 ± 211.57	139.66 ± 82.66	$94.75 \pm 35.99^*$

A computational anatomical atlas should map the structure of the organs across different domains, e.g. different scales of observation, multimodal information sources, across patient populations [81]. Atlases combining multiple sources and a number of individuals provide a powerful tool for describing shapes from a statistical and visual point of view. Figure 5.6 illustrates a population-derived atlas across different domains within the Atlas-R-ASMG deep learning framework (i.e. s/hGCN-ATT) on both LV and liver datasets. The LV atlas provides a statistical average pattern from a large-scale LV population from the CMR image dataset. The liver atlas, on the other hand, is built from small liver datasets that come from multi-modality images (including CT and PET-CT).

The accuracy of the shape matching procedure is evaluated by comparing the R-ASM method with the rigid Registration-based Shape Matching (RSM) proposed in [65], which represents shape as point clouds, and ASM (in Chapter 4). The purpose of this evaluation is to analyze the importance of surface representation of shapes, as well as refinement strategy in attention-based shape matching. To project different-sized actual shapes onto the atlas domain, each method learns a correspondence map from node space $\mathcal{F}(\mathbb{R}^{N_k})$ to $\mathcal{F}(\mathbb{R}^{N_\mu})$. Table 5.1 summarises the accuracy of the test shapes (the 200 LV cases and 28 liver cases test set), in the atlas domain obtained by different methods on LV and liver datasets, where the quality of the obtained shapes is assessed using two distance metrics: Hausdorff distance (HD) and Chamfer distance (CD). The R-ASM method outperforms the RSM and ASM methods across all the metrics. It means the R-ASM framework presents high-quality shapes in the arbitrary atlas domains, by achieving lower mean HD and CD, while the lower standard deviation values indicate its robustness. The refinement improves the quality of the normalised shapes over the sGCN-ATT and hGCN-ATT settings in terms of HD and CD which is visualised in Figures 5.4 and 5.5. The lower HD/CD metrics of the hGCN-ATT compared to the sGCN-ATT reflects the observation that the sGCN-ATT is still susceptible to producing topological errors which can be rectified by refinement procedure (see lavender-coloured shapes in Figures 5.4 and 5.5).

The improvement achieved with refinement over the attention-based shape matching(ASM) is also illustrated in Figures 5.7 and 5.8. The scatter plots show distance metrics HD and CD of ASM and R-ASM models on the LV and liver test cases in two different settings. In both settings on both LV and liver shapes, the minority of samples

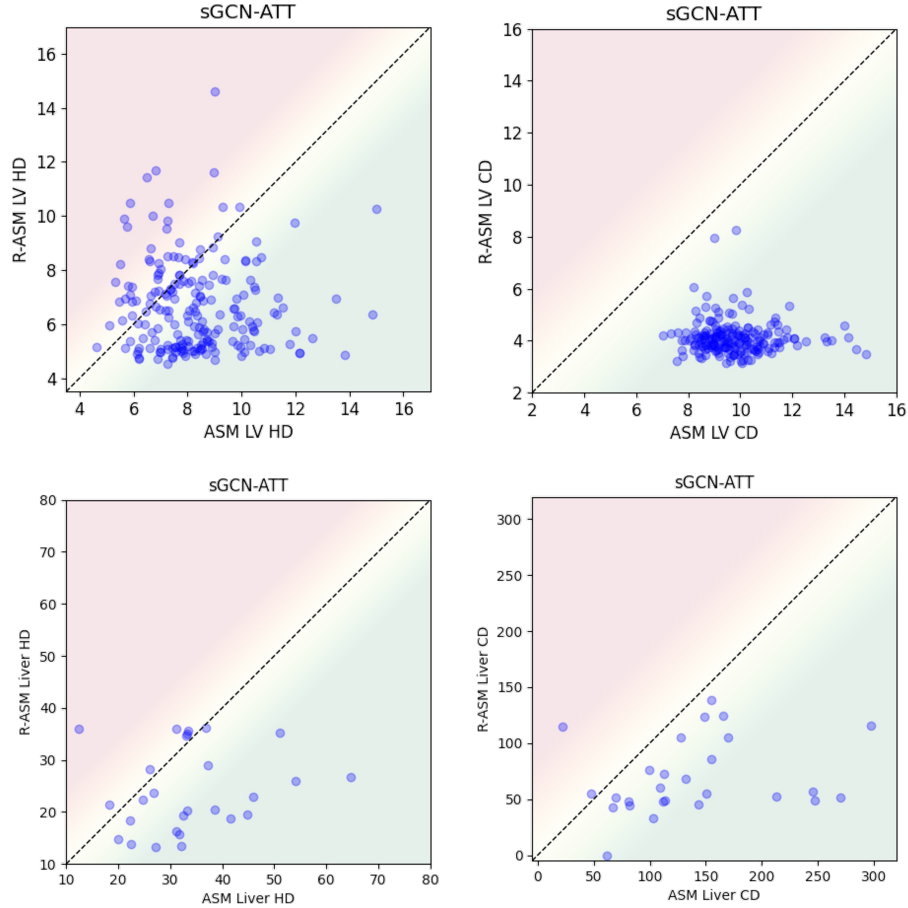


Figure 5.7: Scatter plots showing HD (left) and CD (right) results of attention-based shape matching frameworks in $[mm]$, with "sGCN-ATT" setting, on the 200 LV and 28 liver cases test set, comparing the ASM (x -axis) versus refinable model R-ASM (y -axis). The green/red gradients indicate an increase/decrease in performance with refinement. Refinement generally improves HD and CD of attention-based shape matching results. Degradation is observed for some LV/liver cases in terms of HD, whereas improvements in HD can be significant.

have degradation in the metrics, while outliers are generally corrected, particularly for the metric CD. The higher performance of the R-ASM model compared to ASM is observed across both settings. The improvement is particularly achieved in cases where the inputs to the network are not rich (i.e. sGCN-ATT settings with spatial features inputs in Figure 5.7).

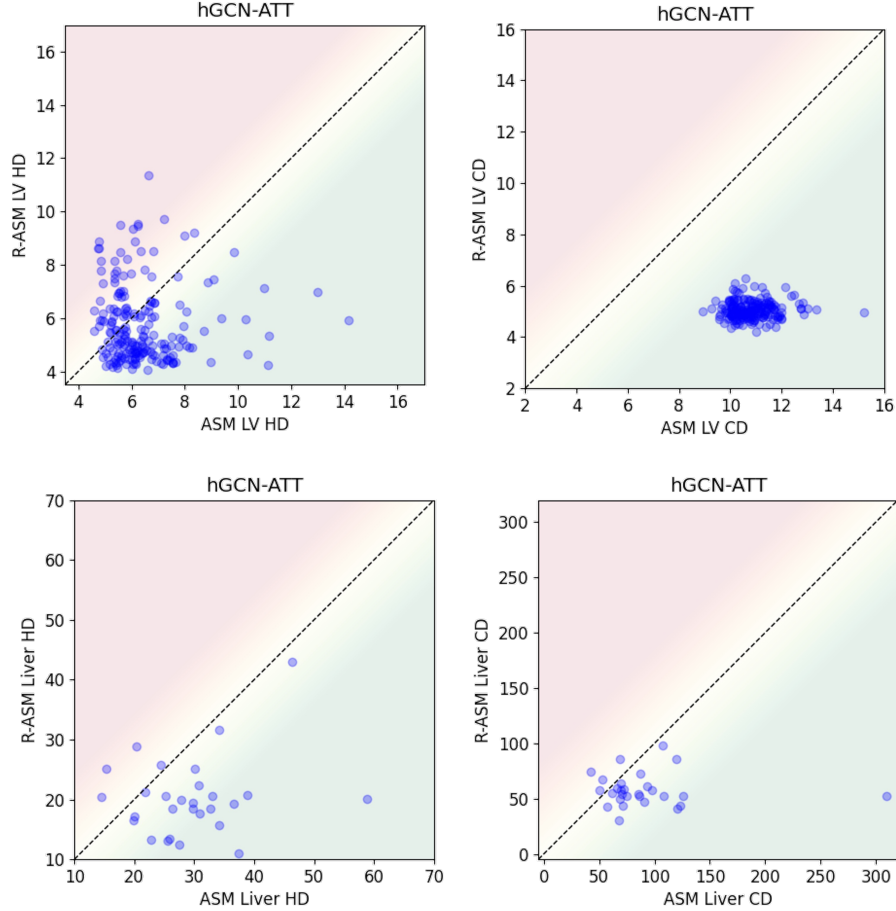


Figure 5.8: Scatter plots showing HD(left) and CD (right) results of attention-based shape matching frameworks in $[mm]$, with "hGCN-ATT" setting, on the 200 LV and 28 liver cases test set, comparing the ASM (x -axis) versus refinable model R-ASM (y -axis). The green/red gradients indicate an increase/decrease in performance with refinement. In the majority of cases, HD and CD of the attention-based shape matching results are improved with refinement. Degradation is observed for some LV/liver cases in terms of HD, whereas improvements in HD can be significant.

Generally, it is observed that high-quality shapes in arbitrary atlas domains are demonstrated by the R-ASM method through achieving lower HD and CD metrics. Furthermore, the values are less spread out (i.e. lower standard deviation), indicating that the R-ASM method is highly robust and is capable of handling a variety of different morphologies.

5.3.2 Evaluation of Generation

The method’s generative capabilities are investigated in terms of generalisation and specificity, demonstrating its ability to synthesise virtual populations of realistic shapes with volumetric measurements in line with actual clinical indices.

5.3.2.1 Generalisability and Specificity

The generalisability of a model is its ability to capture the variability of the seen data (training) and to generalize that to, or explain, the unseen data (testing). Figure 5.9 illustrates generalisation ability of models investigated in this chapter for LV and liver datasets, and compares them with other models. In generative shape models, generalisability can be assessed by evaluating the error in reconstructing unseen test data, thus analyzing the model’s ability to explain unseen shapes and offering insights into the overall variability in shapes. Therefore in Figure 5.9, boxplots present generalisation errors in three metrics: HD , ED and ED^* (in $[mm]$) while ASMG and Atlas-R-ASMG models are implemented in s/hGCN-ATT-VAE settings. It is observed that Atla-R-ASMG(h) models significantly outperform RSMP for all the metrics on both datasets, while the Atla-R-ASMG(s) models significantly outperform the PCA-based model RSMP for the majority of metrics. This is due to the limitation of PCA-based models. These models are prone to overfitting to limited training data, thus not being able to accurately represent anatomies that lie outside of the training distribution, and reconstruction of unseen anatomical structures can subject to significant (large-scale) and subtle (small-scale) variations. Additionally, generalisation error values associated with Atlas-R-ASMG models are consistently lower when compared with those in ASMG models. The Atlas-R-ASMG(h) model significantly outperformed the RSMG(h) model. Hence, learning rich node feature representation and refinable frameworks is beneficial for presenting models with good generalisability. The Atlas-R-ASMG(h) models are able to capture a greater degree of shape variability for each structure (e.g. LV and liver) than afforded by other models and thereby can synthesise more diverse (in terms of shape) virtual anatomical shape populations than the latter.

In order to quantify the anatomical plausibility of synthesised virtual shape populations, specificity errors are employed. The distance between each generated sample in the virtual population and the closest (or most similar) shape in the actual population is measured to calculate these values. Boxplots in Figure 5.10 illustrate the specificity

errors obtained for virtual (LV/liver) populations synthesised using different models. Errors HD , ED , and ED^* are quantified for each structure. Results show that for most metrics, Atlas-R-ASMG significantly outperformed RSMP across the two structures.

For both datasets, the highest specificity (i.e. lower specificity errors) was achieved by Atlas-R-ASMG(h), and its performance was improved compared to ASMG(h). By refining nodal embeddings in an end-to-end framework, rich latent representations can be learned in graph convolutional networks, and the model is able to capture more details in the shape matching and normalisation process. As a result, the VAE generator is then trained with more plausible shapes. Intuitively, this forces a greater degree of plausibility in the shape samples generated from the trained model.

5.3.2.2 Clinical Relevance

A clinical acceptance rate \mathcal{A} is used as an additional metric to assess virtual cohort anatomical plausibility. It is motivated by the need to preserve clinically relevant anatomical volumetric indices in the synthesised cohorts (as compared to the reference actual population).

Analogous to Chapter 4, given actual cohorts G , we assess the clinical relevance of LV/liver virtual cohorts G^* synthesised using the models investigated in this study. To this end, three acceptance rates \mathcal{A}^r , \mathcal{A}^μ and \mathcal{A}^M (defined in Section 3.2.2) are utilised. Acceptance criteria measure how likely it is that synthesised samples in a virtual population with biomarkers (like volume indices) reflect the distribution of the actual biomarker indices. The statistical description of each structure index in the whole population is summarised in Table 3.1. The distribution of LV/liver volume indices is illustrated in Figure 5.11 for the actual samples and those generated using different generative models RSMP, ASMG, and Atlas-R-ASMG. In the figure, s/hGCN-ATT-VAE settings are referred to as s/h and horizontal lines mark the bounds for the different acceptance criteria. As shown in Figure 5.11, comparing the actual to synthetic distributions for volume variables, the Atlas-R-ASMG model (especially for hGCN-ATT-VAE) generates realistic synthetic populations for LV and liver volumes, while capturing sufficient variability. This is consistent with the specificity errors summarised in Figure 5.10. Intuitively, the investigated methods generate distributions of (LV/liver) biomarkers that follow the range defined by \mathcal{A}^r , presenting reasonable shape variability.

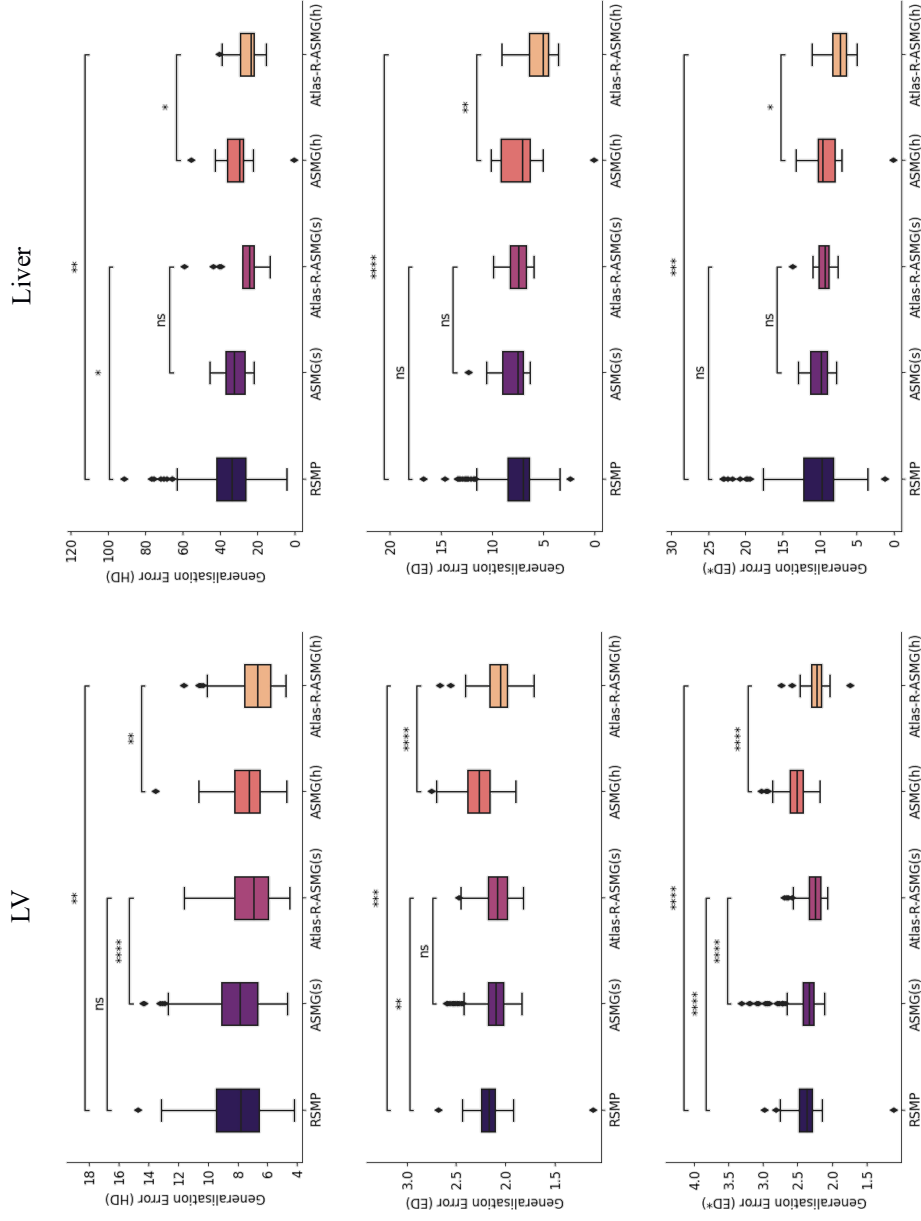


Figure 5.9: Generalisation ability: Boxplots show the generalisation errors (in $[mm]$) for the different models RSMP, ASM, and Atlas-R-ASMG with different settings s/h, where "s" and "h" refer to sGCN-ATT-VAE and hGCN-ATT-VAE respectively and the models' performance are statistically compared. The first and second columns illustrate results on LV and liver datasets, respectively. In each row, HD , ED , and ED^* distances are reported from top to bottom.

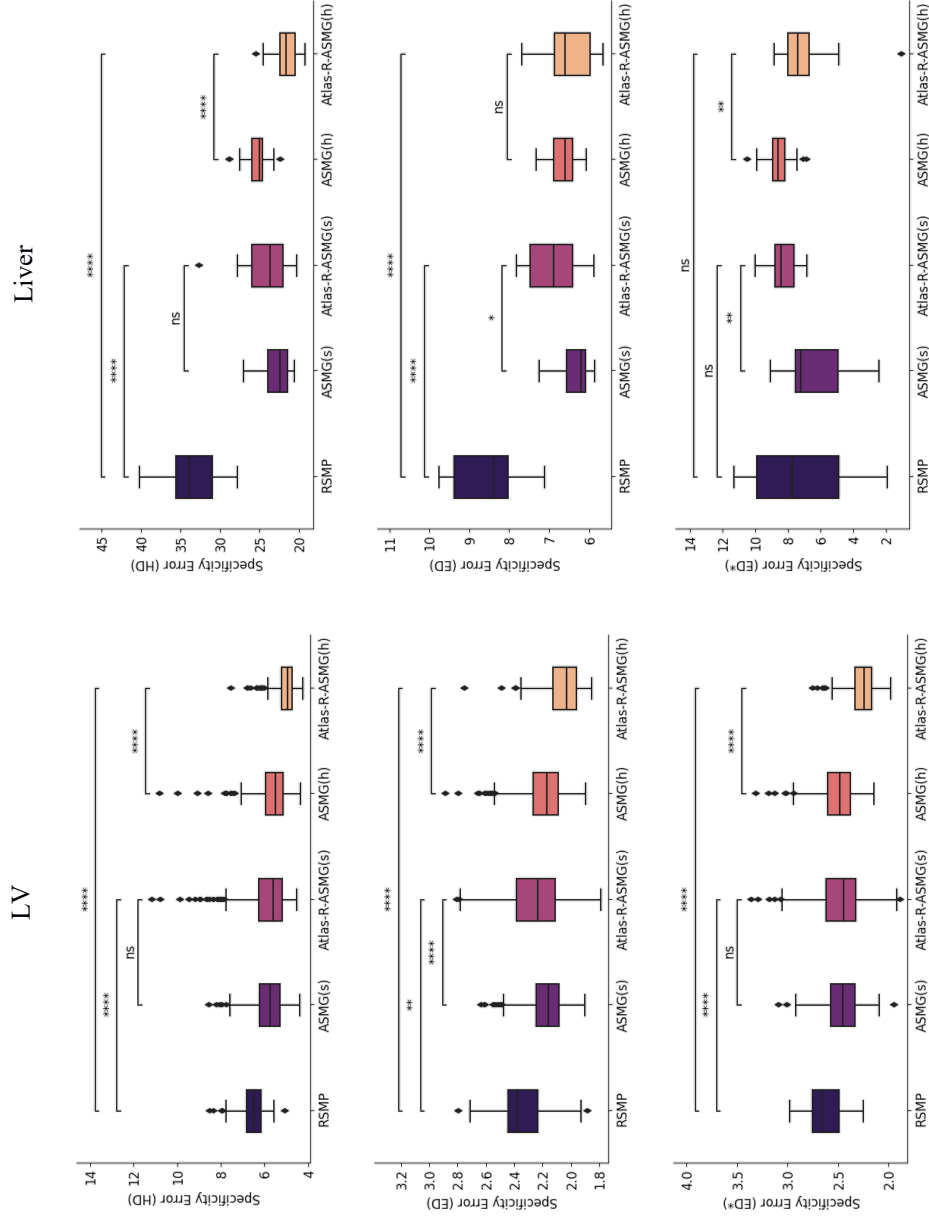


Figure 5.10: Specification ability: Boxplots show the specificity errors (in $[mm]$) for the different models RSMP, ASMG, and Atlas-R-ASMG with different settings s/h, where "s" and "h" refer to sGCN-ATT-VAE and hGCN-ATT-VAE respectively and the models' performance are statistically compared. The first and second columns illustrate results on LV and liver datasets, respectively. In each row, HD , ED , and ED^* distances are reported from top to bottom.

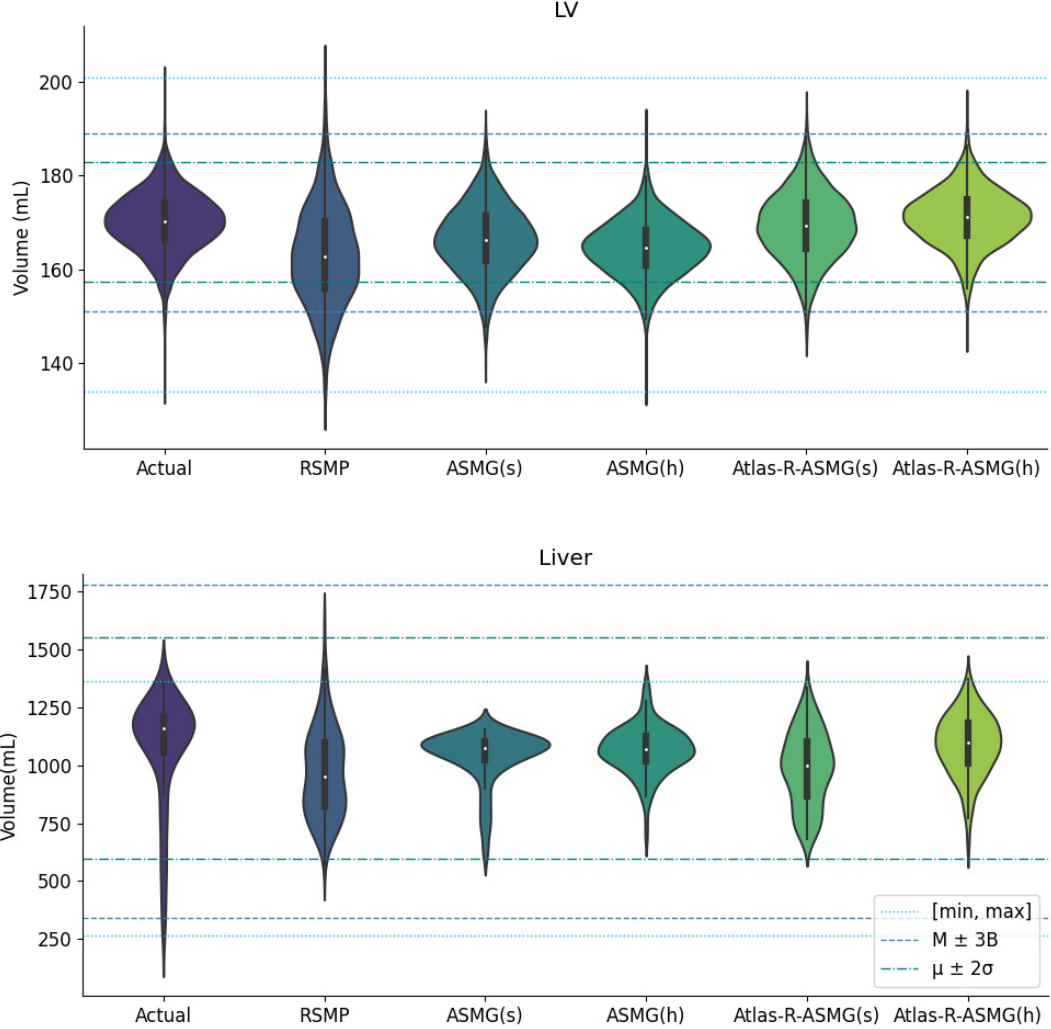


Figure 5.11: Violin plots for the distribution of LV and liver volume indices on the actual samples, alongside with those generated using different methods: RSMP, ASMG(s/h) and Atlas-R-ASMG(s/h). Horizontal lines mark the confidence intervals for the different acceptance criteria.

Figures 5.12 and 5.13 show the anatomy of three synthetic LV and livers within different acceptance intervals. In Figure 5.12, LV shape g_1^* meets \mathcal{A}^r , \mathcal{A}^M and \mathcal{A}^μ , g_2^* meets \mathcal{A}^r and \mathcal{A}^M but not \mathcal{A}^μ and g_3^* only meets \mathcal{A}^r . For liver shapes, in Figure 5.13, g_1^* and g_2^* meets all of the criteria and g_3^* meets \mathcal{A}^M and \mathcal{A}^μ but not \mathcal{A}^r .

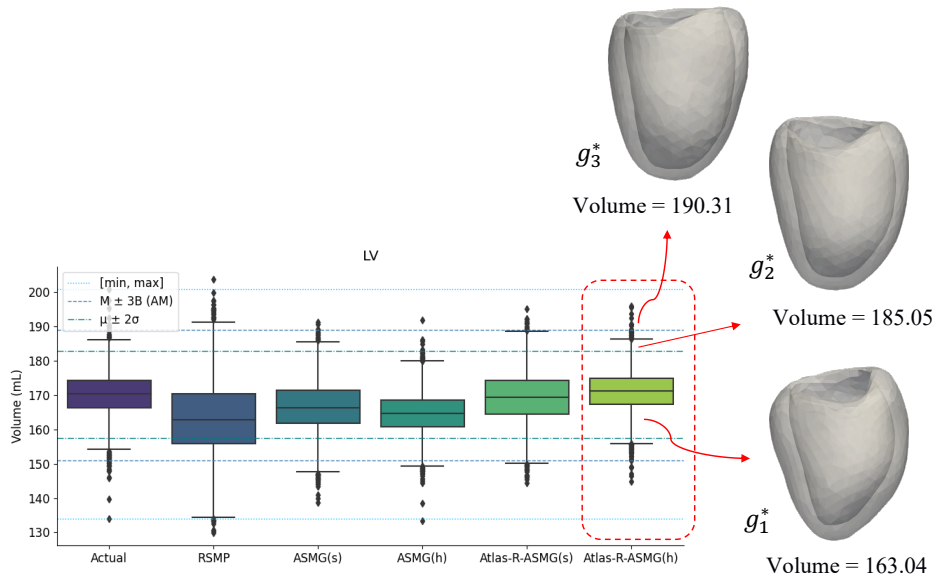


Figure 5.12: Examples of three synthetic LV shapes generated by Atlas-R-ASMG(h) model, with decreasing feasibility of the biomarkers according to the acceptance functions. From left to right: g_1^* is accepted by all of the criteria; g_2^* is only rejected by \mathcal{A}^μ but not by the other acceptance functions; and g_3^* is rejected by \mathcal{A}^μ and \mathcal{A}^M but accepted by \mathcal{A}^r .

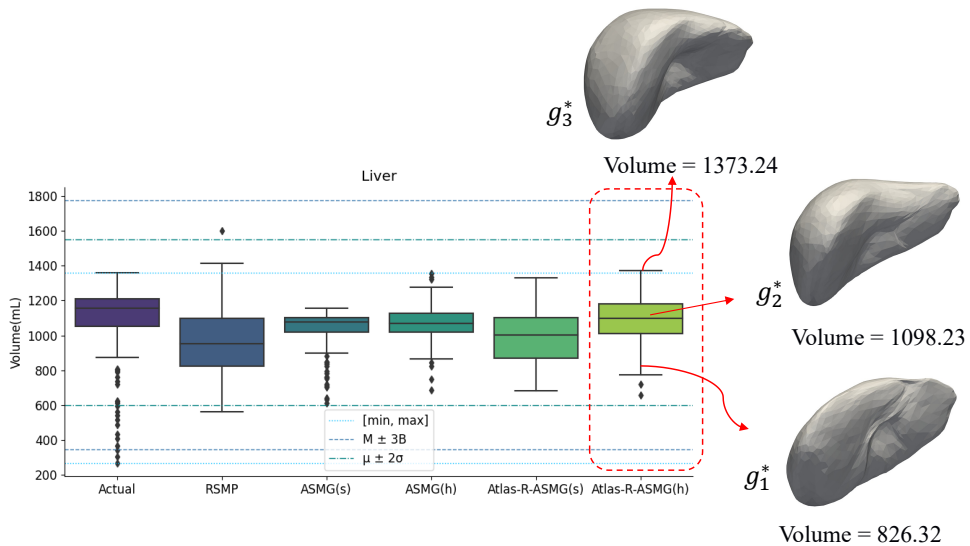


Figure 5.13: Examples of three synthetic LV shapes generated by Atlas-R-ASMG(h) model, with decreasing feasibility of the biomarkers according to the acceptance functions. From left to right: g_1^* and g_2^* are accepted by all of the criteria; g_3^* is only rejected by \mathcal{A}^r but not by the other acceptance functions.

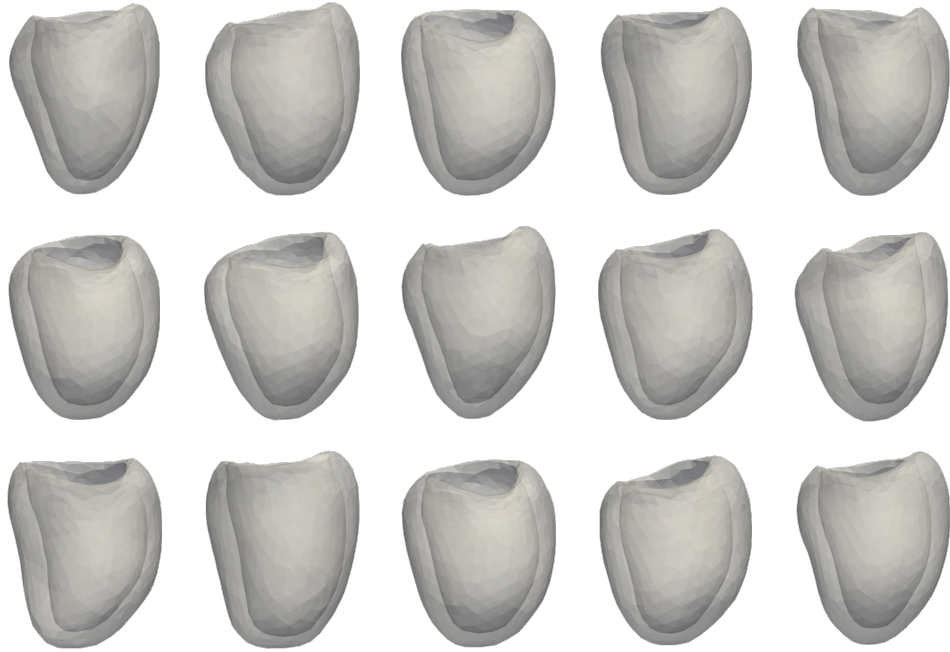
Table 5.2: Clinical acceptance rates \mathcal{A} [in %] achieved by different generative models for LV and liver volumes.

		RSMP	ASMG(s)	ASMG(h)	Atlas-R-ASMG(s)	Atlas-R-ASMG(h)
LV	\mathcal{A}^r	99.75	100	99.98	100	100
	\mathcal{A}^M	86.75	98.10	98.67	99.12	99.65
	\mathcal{A}^μ	65.85	88.65	88.85	91.47	95.85
Liver	\mathcal{A}^r	96.40	100	100	100	99.28
	\mathcal{A}^M	100	100	100	100	100
	\mathcal{A}^μ	98.56	100	100	100	100

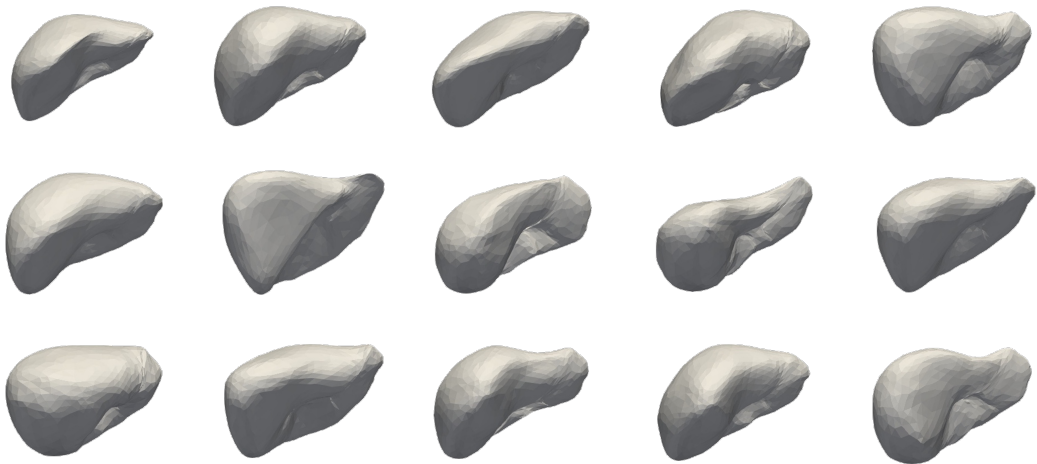
Table 5.2 summarises the acceptance rates \mathcal{A} calculated for LV and liver structures in synthesised virtual cohorts. The clinical acceptance rates of LV/liver virtual cohorts synthesised by the proposed generative models are higher than the statistical shape RSMP model. Furthermore, the values estimated for Atlas-R-ASMG are consistently higher than those obtained from ASMG across the LV volume indices, and there is no huge difference between the values obtained for liver indices.

In conclusion, the results demonstrate that the four models (ASMG(s/h), Atlas-R-ASMG(s/h)) in the proposed generative shape framework better preserves the clinical relevance of anatomical indices (within a 95% confidence interval of the observed values in the actual population) compared to PCA-based models. As PCA-based models (e.g. RSMP) are linear projections of shape data onto a lower-dimensional subspace, they are not capable of capturing non-linear variations in shapes. Thus, models trained using such approaches have limited generalisation capacity and specificity (as shown in Figure 5.9 and Figure 5.10) and correspondingly, synthesised virtual cohorts have limited anatomical plausibility, as evidenced by the results in Table 5.2. The VAE-based graph convolutional generative shape models (e.g. ASMG and Atlas-R-ASMG), on the other hand, are able to capture non-linear variations in shapes, resulting in virtual cohorts with a greater degree of specificity/anatomical plausibility (as shown in Figure 5.10 and Table 5.2) as well as a better generalisation to unseen shapes (see Figure 5.9).

Amongst the VAE-based shape models investigated in this study, the Atlas-R-ASMG(h) generator model allows for refinement of the derived nodal embeddings from



(a)



(b)

Figure 5.14: Examples of virtual (LV/liver) samples generated by the Atlas-R-ASMG generator model.

hybrid representation, in order to optimise the alignment of training shapes on the atlas. This process mitigates disorder in the latent space, which can result in spurious correspondences. Additionally, it improves the accuracy of shape normalisation, preserving the plausibility of shapes. This enables the the Atlas-R-ASMG(h) generator models to generate virtual cohorts with better specificity/anatomical plausibility (refer to Figure 5.10) and provide better fidelity in preserving the distributions of clinically relevant LV/liver indices in the synthesised virtual cohorts relative to the actual population (as shown in Table 5.2).

Figure 5.15 demonstrates a visualisation of the generated (synthetic) samples by the Atlas-R-ASMG. One can observe the generated meshes demonstrate a level of shape variation that closely resembles that of the actual population for both the left ventricle (LV) and liver shapes. Simultaneously, they retain realistic anatomical shapes at an individual level.

5.4 Joint Clustering Generative Shape Model

In medical applications, some organs population (like the liver) show large variations in shape between different individuals. To get a better understanding of the normal and pathological functioning of the shape, it is important to preserve and capture these variations in the shape analysis process. In order to analyze morphological variability, it is possible to cluster shape populations into subgroups with more localized modes of variability. In medical imaging, for instance, this approach may be useful from an application point of view; rather than largely deforming a single mean shape, each subgroup can be associated with a specific disorder, age, gender, etc., with local means representing more natural average anatomy. In addition, in ISCTs, providing realistic precise synthetic shapes can improve shape-based risk assessment and treatment planning. In the development of products such as implants or customized instrumentation, it is crucial to meet the anatomy related morphological needs of a patient as precisely as possible, since otherwise, the outcome of medical intervention may not meet patient expectations.

The proposed ASMG and Atlas-R-ASMG schemes considered a unique atlas for each shape category and relied on regressing all shapes (regardless of morphological variation) on a common atlas where correspondence metrics can be computed. Although the proposed frameworks enable learning from graph populations featuring dis-

5.4 Joint Clustering Generative Shape Model

parate connectivities, the generated meshes have the same connectivity, defined by the atlas. However, when the population of shapes shows large variability in appearance and anatomical structure it becomes arduous to come up with a unique atlas that is representative of all the rest. This certainly can increase distance errors and diminish fidelity.

In this section, the aim is to address these limitations by presenting a "joint clustering generative" shape model and introducing multi-atlas over a specific organ, where atlases have different topological structures. Incorporating the "multi-atlas" shape matching procedure within the framework improves the establishment of the correspondence. This allows for grouping shapes into meaningful clusters based on their similarity measures, making it easier to analyze and understand variation and complex spatial relationships. Furthermore, by incorporating multi-atlas construction with variable topologies, the generative model gains the capability to synthesise virtual shape populations exhibiting diverse topological variations. Therefore, the generative model can generate more morphological variability of shapes, due to the presenting different shape clusters. This is particularly useful in cases where vertex-to-vertex correspondences be challenging or inaccurate due to occlusions or variations in point density. Overall, this model is able to (1) cluster shapes by capturing similarity between them, (2) preserve small details of the input shapes by assigning the best atlas (i.e. best topology) to each observed shape, and (3) generate a multi-resolution virtual population of organ anatomies.

This framework, referred to as the Multi-Atlas Refinable Attention-based Shape Matching Generative model (mAtlas-R-ASMG), represents an extension of the preliminary ideas introduced in Section 5.2. In an atlas-based generative model, the framework provides a map of vertex correspondences between the training shape and the atlas, to structurally normalise shapes (in the shape matching process). To improve the generation quality, a set of atlases is constructed instead of a single atlas for shape normalisation. The results are then combined to create the best agreement shape matching (i.e., structure normalisation). The best agreement can be found by a weighted clustering procedure.

When a set of atlases is available for shape matching, weighted clustering provides a simple and efficient method to identify the best atlas matching (i.e. the best topology for representing the actual shape) and preserve more details in the shape matching pro-

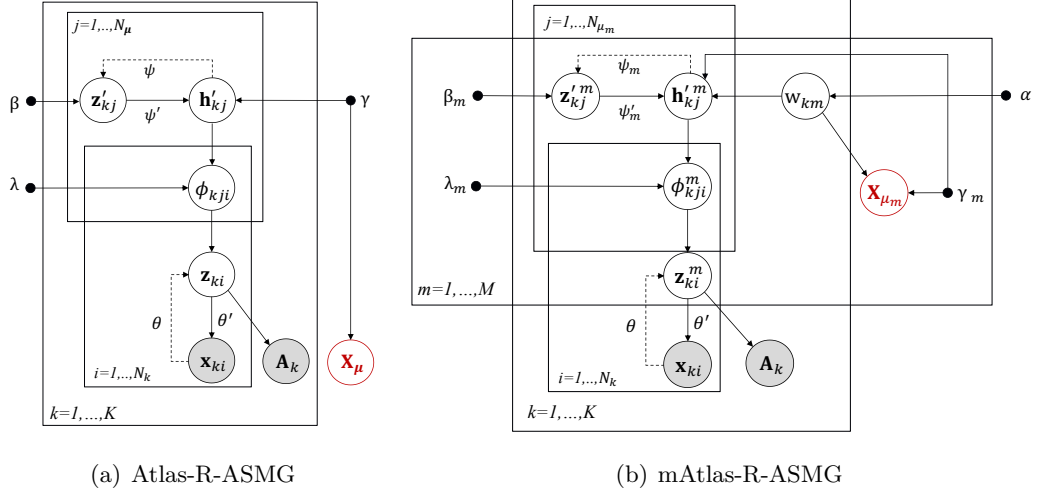


Figure 5.15: Graphical model representation of (a) the Atlas-R-ASMG generative model, and (b) joint clustering generative model (mAtlas-R-ASMG). The observed variables are shown with shaded circles and hyper-parameters are indicated by solid small circles/on the arrows. Other variables and parameters are shown with white circles. Please refer to the text for a detailed explanation.

cess. Each atlas is used for establishing vertex-wise correspondences and then structure normalisation of actual shapes. The selection of the optimal atlas, which represents the most suitable topological structure for describing the shape variation, is determined for each shape through a voting process. This voting is based on a shape-based weight, calculated as a function of the distance between the structurally normalised shape and the atlas.

5.4.1 Clustering Scheme

Assuming an unlabeled shape set $G = \{g_k\}_{k=1}^K$ is projected into M clusters. The shape set $\{g_{\mu_m}\}_{m=1}^M$ illustrates the average shapes (i.e. atlases) in each cluster with matrix representation $\mathbf{X}_{\mu_m} \in \mathbb{R}^{N_{\mu_m} \times 3}$.

Projecting each shape on different atlases, the framework presents M clusters of structurally normalised shape populations where each population presents K shape in the resolution $|V| = N_{\mu_m}$. The shape $g'_k{}^m$ denotes normalised shape k in the cluster m and w_{km} is its associated weight.

5.4 Joint Clustering Generative Shape Model

The weights elements w_{km} are defined as follows:

$$w_{km} = \frac{e^{-\alpha d_{km}}}{\sum_{m=1}^M e^{-\alpha d_{km}}} \quad (5.13)$$

where normalised shape g_k^m is presented by the geometric feature matrix \mathbf{h}_k^m , d_{km} presents the mean squared error between each normalised shape and mean shape in each cluster: $d_{km} = \frac{1}{|N_{\mu_m}|} \sum_j |\mathbf{h}_{kj}^m - \mathbf{x}_{\mu_m j}|^2$; and α is a hyperparameter that controls the de-weighting degree in terms of dissimilarity. Therefore, a higher weight is assigned to a shape that is more similar to the m -th atlas. Shape-wise weights are denoted as vector $w_k \in \mathbb{R}^M$, and weight matrix $W = (w_1, w_2, \dots, w_K)^T$ fulfills the constraint $\sum_{m \in M} w_{km} = 1; \forall k \in K$.

Similar to Equation 5.14, atlas shapes are reconstructed by minimising the following cost function,

$$L_\mu = \frac{1}{2} \sum_{m=1}^M \sum_{k=1}^K \sum_{j=1}^{N_{\mu_m}} w_{km} |\mathbf{h}_{kj}^m - \mathbf{x}_{\mu_m j}|^2 + \frac{\gamma_m}{2} \sum_{j=1}^{N_{\mu_m}} \sum_{q=1}^{N_{\mu_m}} a_{\mu_m pq} |\mathbf{x}_{\mu_m j} - \mathbf{x}_{\mu_m q}|^2 \quad (5.14)$$

where $a_{\mu_m jq}$ are elements of the adjacency matrix \mathbf{A}_{μ_m} (demonstrates the topological structure of m -th cluster). Starting from an initial canonical atlas shape $\mathbf{X}_{\mu_m}^{(0)}$, a new atlas $\mathbf{X}_{\mu_m}^{(i+1)}$ is iteratively computed from $\mathbf{X}_{\mu_m}^{(i)}$ according to

$$\mathbf{x}_{\mu_m j}^{(i+1)} \leftarrow \frac{\sum_k w_{km} \mathbf{h}_{kj}^m + \gamma_m \sum_{q \in N_{\mu_m j}} \mathbf{x}_{\mu_m q}^{(i)}}{\sum_k w_{km} + \gamma_m \sum_{q \in N_{\mu_m j}} a_{\mu_m jq}}; \quad \forall j \in N_{\mu_m}; \forall m \in M \quad (5.15)$$

where $\mathbf{x}_{\mu_m j}$ is a feature vector corresponding to j -th node in the m -th atlas graph and $\gamma_m = N_{\mu_m} / \max(N_{\mu_m j})$. The atlas shape g_{μ_m} is built with a fixed adjacency matrix \mathbf{A}_{μ_m} , ensuring that the vertex positions are optimised while preserving the topology of the atlas. The process of initializing atlases relies on canonical shapes informed by clinical experts' knowledge. Incorporating expert insights adds significant value, allowing the model to present more clinically relevant clusters that capture the expected variations in anatomical organs. This initialization ensures that the atlas is rooted in domain-specific understanding, enhancing the model's ability to generate synthetic shapes that align with the intricacies observed in medical imaging data [124]. The utilization of expert knowledge in the initialization phase contributes to the robustness and clinical relevance of the subsequent multi-clustered learning process.

Consequently, the population can generate M atlas shapes with diverse topologies/cardinalities, showcasing variations in shapes. This approach addresses the limitation of relying on a single atlas shape for normalising real shapes, which may overlook

certain details. By deriving multiple atlas shapes, the method captures a broader range of shape variations, enhancing the accuracy and comprehensiveness of the analysis.

5.4.2 Generative Modeling

To generate virtual populations with variable topology in anatomical structures, a multi-generative model is trained based on different clusters of shape populations. More specifically, for M clusters, our framework presents M generative models where each generative network (designed in β -VAE structure) is trained to derive a pdf from a set of normalised graphs. Therefore, similar to the Section 5.2.2, for the m -th generative model, we optimize the variational lower bound L_{G_m} w.r.t the variational parameters ψ_m, ψ'_m :

$$L_{G_m}(\psi_m, \psi'_m) = \sum_{k=1}^K (\mathbb{E}_{q_{\psi_m}(\mathbf{z}'_k | \mathbf{h}'_k)} [\log p_{\psi'_m}(\mathbf{h}'_k | \mathbf{z}'_k)] - \beta_m D_{KL}[q_{\psi_m}(\mathbf{z}'_k | \mathbf{h}'_k) \parallel p(\mathbf{z}'_k)]); \quad \forall m \in M, \quad (5.16)$$

where \mathbf{h}'_k denotes the geometric features of k -th shape from m -th cluster. The probability of node variations is approximated in the latent space $\mathbf{z}'_k \in \mathbb{R}^L$ via a posterior $q_{\psi_m}(\mathbf{z}'_k | \mathbf{h}'_k)$. By drawing samples from this approximate posterior probability, we estimate the likelihood of the observed population. To generate novel shape samples, we simply sample from the prior normal distribution $p(\mathbf{z}'_k) = N(0, I)$ and use the sample as the input to the decoder, which converts that into a random synthetic mesh having the same connectivity as the m -th atlas (i.e. N_{μ_m}).

Considering Equation 5.13, the shape that is more relevant to cluster m assigns a higher weight, so that it will influence the m -th generative model more than other generative models. Hence, the lost function in Equation 5.16, can be rewritten as

$$L_{G_m} = \frac{1}{2} \sum_{k=1}^K w_{km} \left(\sum_{j=1}^{N_{\mu_m}} \left\| \mathbf{h}'_{k_j}{}^{rec} - \mathbf{h}'_{k_j} \right\|^2 - \beta_m D_{KL}[q_{\psi_m}(\mathbf{z}'_k | \mathbf{h}'_k) \parallel p(\mathbf{z}'_k)] \right) \quad (5.17)$$

where \mathbf{h}'_k and $\mathbf{h}'_k{}^{rec}$ denote the input shape and the its reconstruction respectively. Figure 5.15(b) is a graphical representation of the joint clustering generative model considered in this section, which shows the hypothesised dependencies of the variables. To create a virtual population of shape with "variable topology" in anatomical structures, given the dependencies of the variables, ancestral sampling [125] is used to draw

5.4 Joint Clustering Generative Shape Model

Table 5.3: Shape matching quality in mAtlas-R-ASM model with different number of atlases (i.e. M) on liver shapes. Values show the performance of the model using two distance metrics HD and CD (mean \pm std) in $[mm]$. **Bold** values show a significant difference between the methods with a p-value < 0.001 using the statistical paired t-test.

	$M = 1$	$M = 3$	$M = 5$
HD	20.21 \pm 6.35	18.07 \pm 6.39	17.33 \pm 5.43
CD	58.14 \pm 21.96	33.64 \pm 12.33	33.84 \pm 10.93

3D surface mesh samples from our generative model. Samples are drawn from models of M clusters with L latent dimension.

5.4.3 Experimental Results and Discussion

As discussed earlier in Section 5.4, joint clustering generative models excel in scenarios where large variability exists in appearance and anatomical structure within the shape population. They enhance the generative model’s capability to synthesise virtual shape populations with diverse topological variations. The liver is a complex organ known for exhibiting significant morphological variations among individuals, making it an ideal candidate for shape clustering and generative analysis. Hence, in this section, the liver dataset is employed to evaluate the joint-clustering generative model and explore the promising approach of multi-atlas construction along with its benefits.

For experimental settings, hyperparameter $\alpha = \frac{1}{std(d)}$ which is experimentally found appropriate. Here the vector $d_k \in \mathbb{R}^M$, and distance matrix $d = (d_1, d_2, \dots, d_K)^T$. To build a multi-generative framework, M VAE generative networks share hidden layers. Since earlier results in Section 5.3 demonstrated significant improvements in anatomical validity with hybrid representations of shapes, the hybrid representation is employed in the experimental settings for this section. The joint-clustering generative model combines the power of clustering algorithms with generative modelling techniques to achieve accurate shape matching and domain transformation for shape analysis and generation. By leveraging both clustering and generative modeling, this approach can effectively capture the inherent structures and variations present in the liver dataset. Table 5.3 summarises the accuracy of liver shapes in the shape matching procedure

proposed in the joint-clustering generative model. The quality of the obtained shapes is assessed using two distance metrics: Hausdorff distance (HD) and Chamfer distance (CD). The obtained results show that the multiple reference atlases can help improve the accuracy of regressed shapes in different atlas domains. Multiple atlases positively impact the accuracy, reliability, and robustness of the shape matching process by achieving lower mean and standard deviation Hausdorff and Chamfer distances.

By incorporating information from multiple atlases, the model can consider a wider range of anatomical variations, leading to improved matching and, consequently, generation outcomes. Figure 5.16 demonstrates some examples of obtained results for shape matching using multi-atlas clustering (e.g. $M = 5$). In each part, the first and second rows present shapes and the slice-mask images constructed from them, respectively. To accurately evaluate the shapes in different clusters according to the ground-truth mask, masks from the 3D normalised shapes are extracted in different clusters.¹ The similarity and distance between the extracted mask and their corresponding ground-truth are measured using the Dice Similarity Coefficient (DSC) and Intersection over Union (IoU) metrics. The first column shows the input shape/mask and the rest present normalised shapes/masks in different clusters. The weighted clustering algorithm presented in Section 5.4.1 finds the best atlas (green colour) for deriving vertex-to-vertex correspondences. The following table presents the values of d_k in $[mm]$ and w_k as given in Equation 5.13 for $M = 5$ for shapes. Additionally, it includes the DSC and IoU metrics for masks.

To test accuracy in different clusters, two metrics: DSC and IoU are defined as:

























$$\text{DSC} = 2 \frac{|A \cap B|}{|A| + |B|} \quad (5.18)$$

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \quad (5.19)$$

where A is the mask related to the input shape and B shows the mask constructed from the obtained shape (based on established correspondences within each cluster). The visual inspection of the results in Figure 5.16 shows that, for each input shape, the winning cluster presents a richer representation of liver anatomy. In addition, the numerical results indicate that the winning cluster is characterized by reduced distances

























¹The "vtkPolyDataToImageStencil" class is used to convert polydata (e.g surface mesh) into an image stencil. The ground truth image of the actual shape g is considered as a reference vtkImage to match the polydata shape g'^m with.

5.4 Joint Clustering Generative Shape Model

Input	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Input	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
											
											
$d \downarrow$	33.4221	230.2048	277.8301	167.6661	151.1950	$d \downarrow$	180.6635	106.5243	205.1174	107.6714	247.9251
$w \uparrow$	0.4103	0.1217	0.0907	0.1791	0.1982	$w \uparrow$	0.1764	0.2788	0.1516	0.2768	0.116
DSC \uparrow	0.9677	0.9667	0.9674	0.9633	0.9675	DSC \uparrow	0.9562	0.9680	0.9638	0.9626	0.9601
IoU \uparrow	0.9374	0.9356	0.9369	0.9293	0.9371	IoU \uparrow	0.9161	0.9380	0.9302	0.9280	0.9234













(a) Cluster 1

(b) Cluster 2

Input	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Input	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
											
											
$d \downarrow$	336.5668	263.6840	221.3773	318.8008	281.5244	$d \downarrow$	433.6566	419.4966	653.3057	385.5603	447.4217
$w \uparrow$	0.1403	0.2201	0.2858	0.1566	0.1971	$w \uparrow$	0.2168	0.2366	0.0558	0.2917	0.1991
DSC \uparrow	0.9230	0.9271	0.9514	0.9240	0.9360	DSC \uparrow	0.9405	0.9513	0.9504	0.9540	0.9511
IoU \uparrow	0.8570	0.8642	0.9073	0.8588	0.8795	IoU \uparrow	0.8877	0.9071	0.9056	0.9120	0.9068

(c) Cluster 3

(d) Cluster 4

Input	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
					
					
$d \downarrow$	265.0753	229.6202	447.6724	188.7720	77.4586
$w \uparrow$	0.1359	0.1692	0.0440	0.2178	0.4331
DSC \uparrow	0.9504	0.9587	0.9572	0.9600	0.9637
IoU \uparrow	0.9055	0.9207	0.9179	0.9230	0.9300

(e) Cluster 5

Figure 5.16: Examples of clustering results on liver datasets, using the Multi-atlas clustering method with $M = 5$ clusters, for establishing shape matching. Arrows indicate the direction of metric improvement. Bold numbers are the best for a given metric/parameter.

5.4 Joint Clustering Generative Shape Model

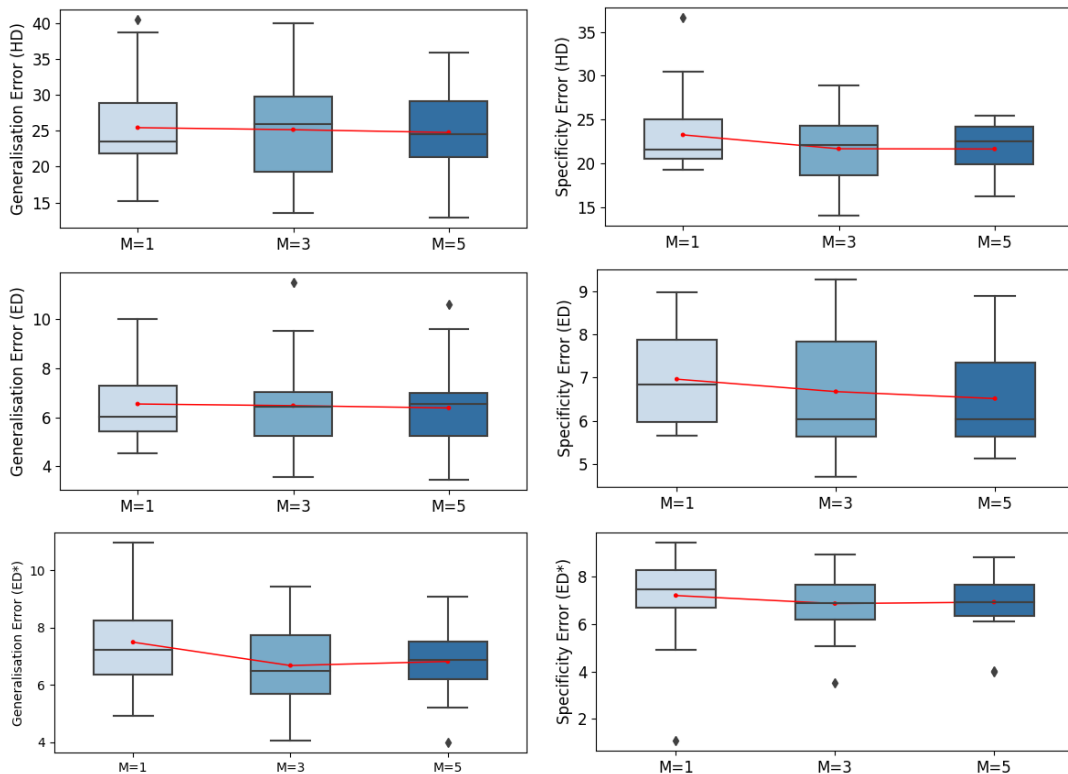


Figure 5.17: Generalisation and Specificity errors (in [mm]) of the mAtlas-R-ASMG model with different values of cluster m . Three distance measures HD , ED and ED^* report the model specificity and generalisation. Boxplots with the line connecting mean values for each group show multi-atlases outperform the single-atlas approaches.

between the input and atlas, as well as higher weights denoted by w . These findings align with the higher DSC and IoU metrics observed.

To assess the impact of shape clustering and multi-atlas construction in generative modelling, Figure 5.17 illustrates generalisation and specificity errors of the clustering generative model (i.e. mAtlas-R-ASMG) with different number of clusters.

The results suggest a preference for a higher number of clusters, with most errors remaining relatively consistent within the range of $M = 3$ to $M = 5$. However, it's important to note that, given the limited size of the liver shape dataset, it may not be sensible to increase the number of clusters to very high values. In Figure 5.17, it can be observed that the clustering generative model improves generalisation and specificity metrics (i.e. lower concurrent specificity and generalisation errors for the majority of

distances). This occurs through various mechanisms. Firstly, it effectively captures complex patterns and relationships within the data by considering multiple clusters. This allows the model to better represent the inherent variability and diversity in the dataset, enabling it to generalise well to unseen shapes and enhance specificity metrics. Additionally, the clustering generative model can incorporate prior knowledge about the class structure (such as initial templates). By leveraging this prior knowledge, the model can guide the clustering and generative modelling process to capture relevant aspects of the data, characteristics or patterns specific. This integration of prior knowledge enhances the model’s generalisation capabilities and improves specificity by identifying class-specific features more accurately.

More importantly, the presented mAtlas-R-ASMG model can be seen as an ensemble of individual clustering and generative models. Each cluster is associated with a specific model and their outputs are combined or weighted to make synthetic shapes. This ensemble approach mitigates errors or biases associated with individual models, leading to improved generalisation and specificity by leveraging the collective knowledge of multiple models.

5.5 Conclusion

An end-to-end unsupervised generative framework named Atlas Refinable Attention-based Shape Matching and Generation (Atlas-R-ASMG) was introduced in this chapter. The objective of this framework was to jointly learn accurate refinable shape matching and generation on 3D surface mesh data while simultaneously constructing a population-derived atlas. The Atlas-R-ASMG network incorporates attention mechanisms to selectively attend to relevant regions of the input data, thereby facilitating precise shape matching. By leveraging attention mechanisms and a refinement procedure, the network is able to effectively establish refined shape matching within the 3D surface mesh data, capturing detailed correspondences between different instances. Furthermore, the framework integrates a generative component, enabling it to generate new shapes that adhere to the learned distribution. This generative capability proves valuable for tasks such as synthesising new shapes that conform to the population’s learned characteristics.

In the learning process, the Atlas-R-ASMG framework constructs a population-derived atlas. This atlas serves as a comprehensive reference template, encapsulating

the common features and variations prevalent within the population of 3D surface mesh data. The construction of a population-derived atlas specific to the dataset under analysis enables the framework to effectively handle the inherent diversity and complexity of the data, thereby facilitating more accurate shape matching and generation.

The ability of the model to extend as a joint clustering generative model (called mAtlas-R-ASMG) holds numerous advantages. It allows for comprehensive analysis and understanding of shapes by capturing both local patterns within clusters and global shape population distribution. Additionally, it facilitates shape synthesis and augmentation by generating new samples with "variable topologies" that closely resemble the original population. Multi-atlas construction and clustering allow the model to learn matching with diverse atlases, enabling it to handle a wide range of variations and achieve more robust correspondences compared to single-atlas approaches. This makes the model particularly valuable when dealing with datasets, like the liver dataset, that exhibit significant morphological variations, including variations in shape and size. Multi-atlas construction inherently addresses uncertainty by taking into account multiple potential matches (i.e. correspondences). This can be particularly beneficial for complex shapes, as it allows the model to handle cases with regions that are challenging to find correspondences accurately. By using clustering shapes into multiple atlases, the model can provide more reliable and robust results.

Training a joint-clustering generative model using multi-atlas construction can enhance the generalisation capabilities of the model. The inclusion of multiple atlases provides a richer representation of shape anatomy, enabling the model to generalise well to unseen shapes and adapt to new patient data. In conclusion, the framework possessed the ability to generate synthetic shapes at variable topology in anatomical structures, realistically replicating shapes from training data that had different topologies.

CHAPTER 6

Conclusions

6.1 Summary and Achievements

This thesis focused on developing an unsupervised probabilistic deep learning framework to cope with the challenges associated with shape matching and generation, presenting a virtual population of anatomical shapes. In particular, this study presented algorithms and methods for generating virtual anatomical shapes resembling actual data. The main contributions of the thesis are outlined in the following paragraphs.

As "graphs" represented shapes in this study, first, Chapter 2 focused on exploring the field of geometric deep learning, which extends deep neural networks to non-Euclidean domains like graphs. The chapter provided a comprehensive review of two categories of geometric deep learning: Spectral-based and Spatial-based methods. The distinctions between spectral and spatial graph convolution methods are emphasized and utilised in this study. The chapter also examined generative shape modelling, including establishing correspondences and generating shapes. It focused on the limitations of point-based methods (e.g. point-set registration) and graph matching for establishing correspondences. A wide range of research on generating medical and non-medical shapes was reviewed, concluding the existing works on shape generative modeling using graphs are valuable but require supervision and specific conditions, such as consistent connectivity and fixed vertex cardinality in training datasets.

Chapter 3 presented a baseline generative model, which is a PCA-based statistical shape model used in point-set registration-based shape matching methods to establish point-to-point correspondences. The chapter highlights the limitations of rigid registration methods, which primarily rely on rigid transformations and struggle to capture non-linear deformations and complex shape variations. Additionally, the drawbacks of PCA-based Statistical Shape Models are examined, including the linearity assumption, limited representational capacity, inability to handle out-of-sample variations, and the challenge of capturing shape correlations. To address these limitations, deep learning-based approaches for shape matching and generation are proposed in subsequent chapters, aiming to capture complex deformations, non-linear shape variations, and improve the representation capabilities compared to PCA-based SSMs.

In Chapter 4, a novel unsupervised deep generative model was proposed for generating virtual anatomical shapes. The aim was to address methodological challenges caused by the lack of consistent tensor-like representation across training shapes. A graph-based generative model was developed to generate realistic synthetic shapes from

graph datasets with variable topologies and without vertex-to-vertex correspondences. Unlike existing shape generative modeling approaches, which require fixed mesh/graph topology (i.e. mesh triangle connectivity should be identical) and supervised vertex correspondences, the proposed framework has the capability to handle variable topologies within the training shape dataset and establishes a learnable set of shape correspondences in the latent space through an unsupervised deep learning approach. The process involved a forward-backward regression of shape datasets on the template model, known as the atlas shape, within an Expectation-Maximisation-like framework. This method simultaneously determined the atlas shape and the vertex-wise correspondence functions necessary to warp the atlas into each observed shape. The convergence of the iterative procedure resulted in structurally normalised meshes, which are then used to train a variational autoencoder for data generation.

By incorporating the spatial-based graph convolutional networks that consider the connectivities between vertices, the proposed model exhibited superior performance in terms of anatomical validity and shape correspondences, surpassing point-based registration approaches. The study demonstrated the suitability and applicability of the proposed generative model in conducting ISCTs, with virtual cohorts, through a comparative analysis. The versatility and performance of the generative model were demonstrated through applications to datasets with wide variability and limited data. However, the chapter also acknowledged limitations such as the need for refinement strategies and the use of a single atlas for all shape categories, which could affect fidelity and distance errors.

Finally, Chapter 5 introduced an end-to-end unsupervised generative framework, the Atlas Refinable Attention-based Shape Matching and Generation network (Atlas-R-ASMG). The proposed model, combining refinable shape matching, generative modeling, and population-derived atlas construction, contributed to the advancement of understanding, analyzing, and synthesising 3D surface mesh data. This end-to-end framework aimed to learn accurate refinable shape matching and generation on 3D surface mesh data while constructing a population-derived atlas. By incorporating attention mechanisms and a refinement strategy, the network selectively attends to relevant regions, enabling precise learnable shape correspondences and capturing detailed correspondences. Additionally, the framework included a generative component, allowing it to generate new shapes adhering to the learned distribution. The construction

of a population-derived atlas facilitates handling the diversity and complexity of the data, leading to more accurate shape matching and generation.

The identified merits of the proposed generative models in this study suggested that for conducting in-silico trials using synthesised virtual cohorts, certain generalisations can be made regarding their suitability in different scenarios. Specifically, utilising hybrid features in the refinement strategy and the virtual anatomical cohorts derived from them proves to be more suitable in scenarios where greater statistical fidelity is needed for the enrolled virtual patients, particularly in terms of relevant anatomical phenotypes.

The model’s ability to extend as a joint clustering generative model (mAtlas-R-ASMG) offers advantages such as comprehensive shape analysis, and synthesis, as well as handling variations and achieving robust correspondences. Multi-atlas construction and clustering enhance the model’s generalisation and specificity capabilities and address uncertainty, particularly beneficial for complex shapes. Training a joint-clustering generative model using multi-atlas construction improves the model’s ability to generalise to unseen shapes and adapt to new patient data. This novel generative framework was designed to handle variable topology in anatomical structures across patients/input samples and successfully synthesised anatomically plausible virtual populations with “diverse topologies”. These capabilities broaden the potential applications of the approach in computational medicine and make it well-suited for in-silico trials. In various medical scenarios, it becomes necessary to model organ/objects with variable topology. One such example is osteoarthritis (OA), which leads to the erosion of articulating cartilage in load-bearing joints, resulting in changes in the object’s topology. To effectively model such objects, the shape model must be flexible enough to accommodate these topological changes, which can be provided by the mAtlas-R-ASMG model.

In the exploration of relevant literature within the field of generative modelling, this study has rigorously evaluated the proposed generative model using established metrics. Drawing from the comprehensive landscape of generative model assessment, we have employed a multifaceted approach, considering metrics that encompass fidelity, diversity, and generalization. Specifically, the evaluation framework incorporates commonly utilized metrics such as specificity and generalization, crucial for synthesizing anatomical shapes. Additionally, distance metrics have been applied to measure dissimilarity, offering insights into how well the generated shapes align with real-world

data. Symmetric metrics have been instrumental in assessing the model’s ability to capture inherent symmetry in anatomical structures. Furthermore, the utilization of statistical metrics contributes to presenting the clinical relevance of the synthesized population. This holistic evaluation strategy ensures a nuanced understanding of the generative model’s performance in diverse dimensions.

6.2 Limitations and Future Research Directions

Although the proposed approach facilitates synthesis of anatomical virtual cohorts using inconsistent training data, there are some limitations and potentials that give rise to interesting future research directions.

The framework has been evaluated using two different datasets of anatomical shapes, with both large and limited numbers of patients, consisting of normal and pathological cases. The LV shape dataset derived from UKB CMR images demonstrates homogeneous cohorts. The UKB CMR dataset lacks the complete range of variability in image quality and CMR protocols found in diverse clinical environments. Additionally, the dataset primarily consists of data from a healthy population within a specific age group. While the data includes some pathological cases, it may not capture the full spectrum of pathological patterns present in other clinical settings that are not currently represented by the UKB cohort. Furthermore, the versatility of the model can be evaluated by applying it to LV shape datasets derived from various image data sources, including heterogeneous cohorts.

Moreover, the model can be tested on different anatomical shapes that exhibit diverse variables in their shapes. Exploring the model’s performance and adaptability across these different scenarios would provide valuable insights and contribute to its comprehensive evaluation and potential enhancements. This opens up exciting avenues for future research in assessing the model’s effectiveness in handling shape variations across multiple datasets and organ types with varying characteristics.

Despite the use of comprehensive metrics in this study, evaluating generative modeling for anatomical shapes proves challenging due to the intricate nature of anatomical structures. This complexity arises from the need to balance multiple factors such as fidelity to real-world shapes, diversity in generated samples, and generalization across different anatomies. Additionally, assessing the clinical relevance and utility of the synthesized shapes adds another layer of complexity to the evaluation process. The

6.2 Limitations and Future Research Directions

challenge lies in developing metrics that can effectively capture these aspects and align with the specific requirements of medical applications. However, there are exciting opportunities for future work to evaluate the model more comprehensively from various perspectives and in diverse applications.

The proposed generative model served as a unified framework for constructing a virtual population of single structural shape anatomies. In the case of synthesising multi-part anatomies, like biventricular, two independent generative models can be utilised to generate plausible virtual left ventricle (LV) and right ventricle (RV) structures. However, an important concern that needs to be addressed is ensuring the correct alignment of the synthesised LV and RV shapes. Specifically, it is crucial to avoid topological errors such as gaps or intersections at the interface between the two structures. These types of errors would constitute anatomically implausible variations in shape and must be carefully addressed during the synthesis process. Such issues may be addressed by incorporating supplementary geometric and/or topological constraints into the learning process, which can be explored in future work.

Hence, a potential research direction involves extending the framework as a generative shape compositional framework to synthesise multi-part structures, such as biventricular or whole heart anatomies. This expansion would enable the framework to generate complex anatomical structures composed of multiple interconnected parts.

APPENDIX A

Appendix

A.1 Shape smoothing using Laplacian over \mathbf{h}_k

Optimization of L_{11} in Equation 4.4 written as

$$\langle \partial L_{11}, \partial \mathbf{h}'_k \rangle = \sum_{k=1}^K \sum_{j=1}^{N_\mu} |\mathbf{h}'_{kj} - \mathbf{h}_{kj}| + \gamma \sum_{j=1}^{N_\mu} \sum_{q=1}^{N_\mu} a_{jq} |\mathbf{h}'_{kj} - \mathbf{h}'_{kq}| = 0 \quad (\text{A.1})$$

$$= \mathbf{h}'_{kj} - \mathbf{h}_{kj} + \gamma \sum_q a_{jq} \mathbf{h}'_{kj} - \gamma \sum_q a_{jq} \mathbf{h}'_{kq} = 0 \quad (\text{A.2})$$

and;

$$\mathbf{h}'_{kj} + \gamma \sum_q a_{jq} \mathbf{h}'_{kj} = \mathbf{h}_{kj} + \gamma \sum_q a_{jq} \mathbf{h}'_{kq} \quad (\text{A.3})$$

Therefore, Equation 4.4 can be iteratively optimised w.r.t. the \mathbf{h}'_{kj} resulting in

$$\mathbf{h}'_{kj} \stackrel{(i+1)}{\leftarrow} \frac{\mathbf{h}_{kj} + \gamma \sum_{q \in \mathcal{N}_{kj}} \mathbf{h}'_{kq} \stackrel{(i)}{}}{1 + \gamma \sum_{q \in \mathcal{N}_{kj}} a_{jq}} \quad (\text{A.4})$$

A.2 Optimising the template shape in a closed form solution

Optimization of L_{12} in Equation 4.6 written as

$$\langle \partial L_{12}, \partial \mathbf{X}_\mu \rangle = \sum_{k=1}^K \sum_{j=1}^{N_\mu} (|\mathbf{h}'_{kj} - \mathbf{x}_{\mu j}| + \gamma \sum_{j=1}^{N_\mu} \sum_{q=1}^{N_\mu} a_{jq} |\mathbf{x}_{\mu q} - \mathbf{x}_{\mu j}|) = 0 \quad (\text{A.5})$$

$$= \sum_{k=1}^K (|\mathbf{h}'_{kj} - \mathbf{x}_{\mu j}| + \gamma \sum_{q=1}^{N_\mu} a_{jq} |\mathbf{x}_{\mu q} - \mathbf{x}_{\mu j}|) = 0 \quad (\text{A.6})$$

Considering a degree matrix $D_{ii} = K + \gamma \sum_{q \in \mathcal{N}_i} a_{iq}$ and $A_{ij} = \gamma a_{ij}$, Equation A.5 can be written as $[\mathbf{D} - \mathbf{A}]\mathbf{X}_\mu = \mathbf{h}'$ and

$$\mathbf{X}_\mu = [\mathbf{D} - \mathbf{A}]^{-1} \mathbf{h}' \quad (\text{A.7})$$

where $\mathbf{h}' = (\mathbf{h}'_j^T)$, $\mathbf{X}_\mu = (\mathbf{x}_{\mu j}^T)$, and $\mathbf{h}'_j = \sum_{k=1}^K h'_{kj}$; $j = \{1, \dots, N_\mu\}$.

REFERENCES

- [1] E. Abadi, W. P. Segars, B. M. Tsui, P. E. Kinahan, N. Bottenus, A. F. Frangi, A. Maidment, J. Lo, and E. Samei, “Virtual clinical trials in medical imaging: a review,” *Journal of Medical Imaging*, vol. 7, no. 4, p. 042805, 2020.
- [2] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, “A comprehensive survey on graph neural networks,” *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.
- [3] N. Verma, E. Boyer, and J. Verbeek, “Feastnet: Feature-steered graph convolutions for 3d shape analysis,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2598–2606, 2018.
- [4] K. Torres, G. Staśkiewicz, M. Śnieżyński, A. Drop, and R. Maciejewski, “Application of rapid prototyping techniques for modelling of anatomical structures in medical training and education,” *Folia morphologica*, vol. 70, no. 1, pp. 1–4, 2011.
- [5] N. Archip, R. Rohling, V. Dessenne, P.-J. Erard, and L. P. Nolte, “Anatomical structure modeling from medical images,” *Computer Methods and Programs in Biomedicine*, vol. 82, no. 3, pp. 203–215, 2006.
- [6] M. Goitein and M. Abrams, “Multi-dimensional treatment planning: I. delineation of anatomy,” *International Journal of Radiation Oncology* Biology* Physics*, vol. 9, no. 6, pp. 777–787, 1983.
- [7] P. Vávra, J. Roman, P. Zonča, P. Ihnát, M. Němec, J. Kumar, N. Habib, and A. El-Gendi, “Recent development of augmented reality in surgery: a review,” *Journal of healthcare engineering*, vol. 2017, 2017.

-
- [8] B. Ng, M. Toews, S. Durrleman, and Y. Shi, "Shape analysis for brain structures," *Shape Analysis in Medical Image Analysis*, pp. 3–49, 2014.
- [9] M. Soufi, Y. Otake, M. Hori, K. Moriguchi, Y. Imai, Y. Sawai, T. Ota, N. Tomiyama, and Y. Sato, "Liver shape analysis using partial least squares regression-based statistical shape model: application for understanding and staging of liver fibrosis," *International Journal of Computer Assisted Radiology and Surgery*, vol. 14, no. 12, pp. 2083–2093, 2019.
- [10] C. Biffi, J. J. Cerrolaza, G. Tarroni, W. Bai, A. De Marvao, O. Oktay, C. Ledig, L. Le Folgoc, K. Kamnitsas, G. Doumou, *et al.*, "Explainable anatomical shape analysis through deep hierarchical generative models," *IEEE transactions on medical imaging*, vol. 39, no. 6, pp. 2088–2099, 2020.
- [11] H. Fujita, "Ai-based computer-aided diagnosis (ai-cad): the latest review to read first," *Radiological physics and technology*, vol. 13, no. 1, pp. 6–19, 2020.
- [12] L. R. Wingfield, C. Ceresa, S. Thorogood, J. Fleuriot, and S. Knight, "Using artificial intelligence for predicting survival of individual grafts in liver transplantation: a systematic review," *Liver Transplantation*, vol. 26, no. 7, pp. 922–934, 2020.
- [13] A. Lin, M. Kolossváry, M. Motwani, I. Išgum, P. Maurovich-Horvat, P. J. Slomka, and D. Dey, "Artificial intelligence in cardiovascular imaging for risk stratification in coronary artery disease," *Radiology: Cardiothoracic Imaging*, vol. 3, no. 1, p. e200512, 2021.
- [14] R. Wang, Z. Jiao, L. Yang, J. W. Choi, Z. Xiong, K. Halsey, T. M. L. Tran, I. Pan, S. A. Collins, X. Feng, *et al.*, "Artificial intelligence for prediction of covid-19 progression using ct imaging and clinical data," *European radiology*, vol. 32, pp. 205–212, 2022.
- [15] J. Kim, M. Chae, H.-J. Chang, Y.-A. Kim, and E. Park, "Predicting cardiac arrest and respiratory failure using feasible artificial intelligence with simple trajectories of patient data," *Journal of clinical medicine*, vol. 8, no. 9, p. 1336, 2019.
- [16] A. Tucker, Z. Wang, Y. Rotalinti, and P. Myles, "Generating high-fidelity syn-

- thetic patient data for assessing machine learning healthcare software,” *NPJ digital medicine*, vol. 3, no. 1, pp. 1–13, 2020.
- [17] R. J. Chen, M. Y. Lu, T. Y. Chen, D. F. Williamson, and F. Mahmood, “Synthetic data in machine learning for medicine and healthcare,” *Nature Biomedical Engineering*, vol. 5, no. 6, pp. 493–497, 2021.
- [18] A. Arora and A. Arora, “Synthetic patient data in health care: a widening legal loophole,” *The Lancet*, vol. 399, no. 10335, pp. 1601–1602, 2022.
- [19] M. Viceconti, A. Henney, and E. Morley-Fletcher, “In silico clinical trials: how computer simulation will transform the biomedical industry,” *International Journal of Clinical Trials*, vol. 3, no. 2, pp. 37–46, 2016.
- [20] F. Pappalardo, G. Russo, F. M. Tshinanu, and M. Viceconti, “In silico clinical trials: concepts and early adoptions,” *Briefings in bioinformatics*, vol. 20, no. 5, pp. 1699–1708, 2019.
- [21] F. Shen, B. Chen, Q. Guo, Y. Qi, and Y. Shen, “Augmented reality patient-specific reconstruction plate design for pelvic and acetabular fracture surgery,” *International journal of computer assisted radiology and surgery*, vol. 8, no. 2, pp. 169–179, 2013.
- [22] S. Barone, A. Paoli, A. V. Razonale, and R. Savignano, “3d reconstruction of individual tooth shapes by integrating dental cad templates and patient-specific anatomy,” in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 46285, p. V01AT02A073, American Society of Mechanical Engineers, 2014.
- [23] O. Kodym, M. Španěl, and A. Herout, “Skull shape reconstruction using cascaded convolutional networks,” *Computers in Biology and Medicine*, vol. 123, p. 103886, 2020.
- [24] A. Lopez-Perez, R. Sebastian, and J. M. Ferrero, “Three-dimensional cardiac computational modelling: methods, features and applications,” *Biomedical engineering online*, vol. 14, no. 1, pp. 1–31, 2015.
- [25] A. Bratt, J. Kim, M. Pollie, A. N. Beecy, N. H. Tehrani, N. Codella, R. Perez-Johnston, M. C. Palumbo, J. Alakbarli, W. Colizza, *et al.*, “Machine learning

- derived segmentation of phase velocity encoded cardiovascular magnetic resonance for fully automated aortic flow quantification,” *Journal of Cardiovascular Magnetic Resonance*, vol. 21, no. 1, pp. 1–11, 2019.
- [26] K. Gillette, M. A. Gsell, A. J. Prassl, E. Karabelas, U. Reiter, G. Reiter, T. Grandits, C. Payer, D. Štern, M. Urschler, *et al.*, “A framework for the generation of digital twins of cardiac electrophysiology from clinical 12-leads ecgs,” *Medical Image Analysis*, vol. 71, p. 102080, 2021.
- [27] J. Corral-Acero, F. Margara, M. Marciniak, C. Rodero, F. Loncaric, Y. Feng, A. Gilbert, J. F. Fernandes, H. A. Bukhari, A. Wajdan, *et al.*, “The digital twin™ to enable the vision of precision cardiology,” *European heart journal*, vol. 41, no. 48, pp. 4556–4564, 2020.
- [28] C. Rodero, M. Strocchi, M. Marciniak, S. Longobardi, J. Whitaker, M. D. O Neill, K. Gillette, C. Augustin, G. Plank, E. J. Vigmond, *et al.*, “Linking statistical shape models and simulated function in the healthy adult human heart,” *PLoS computational biology*, vol. 17, no. 4, p. e1008851, 2021.
- [29] O. J. Britton, A. Bueno-Orovio, K. Van Ammel, H. R. Lu, R. Towart, D. J. Gallacher, and B. Rodriguez, “Experimentally calibrated population of models predicts and explains intersubject variability in cardiac cellular electrophysiology,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 23, pp. E2098–E2105, 2013.
- [30] D. G. Kendall, “A survey of the statistical theory of shape,” *Statistical Science*, vol. 4, no. 2, pp. 87–99, 1989.
- [31] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, “Active shape models—their training and application,” *Computer vision and image understanding*, vol. 61, no. 1, pp. 38–59, 1995.
- [32] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, “DeepSDF: Learning continuous signed distance functions for shape representation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 165–174, 2019.

-
- [33] A. Gooya, K. Lekadir, I. Castro-Mateos, J. M. Pozo, and A. F. Frangi, “Mixture of probabilistic principal component analyzers for shapes from point sets,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 891–904, 2017.
- [34] C. Nash and C. K. Williams, “The shape variational autoencoder: A deep generative model of part-segmented 3d objects,” in *Computer Graphics Forum*, vol. 36, pp. 1–12, Wiley Online Library, 2017.
- [35] A.-L. Barabási and R. Albert, “Emergence of scaling in random networks,” *science*, vol. 286, no. 5439, pp. 509–512, 1999.
- [36] R. Albert and A.-L. Barabási, “Topology of evolving networks: local events and universality,” *Physical review letters*, vol. 85, no. 24, p. 5234, 2000.
- [37] V. Batagelj and U. Brandes, “Efficient generation of large random networks,” *Physical Review E*, vol. 71, no. 3, p. 036113, 2005.
- [38] U. Meyer and M. Penschuck, “Generating massive scale-free networks under resource constraints,” in *2016 Proceedings of the Eighteenth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pp. 39–52, SIAM, 2016.
- [39] D. Chakrabarti, Y. Zhan, and C. Faloutsos, “R-mat: A recursive model for graph mining,” in *Proceedings of the 2004 SIAM International Conference on Data Mining*, pp. 442–446, SIAM, 2004.
- [40] J. Leskovec, D. Chakrabarti, J. Kleinberg, and C. Faloutsos, “Realistic, mathematically tractable graph generation and evolution, using kronecker multiplication,” in *European conference on principles of data mining and knowledge discovery*, pp. 133–145, Springer, 2005.
- [41] O. Van Kaick, H. Zhang, G. Hamarneh, and D. Cohen-Or, “A survey on shape correspondence,” in *Computer graphics forum*, vol. 30, pp. 1681–1707, Wiley Online Library, 2011.
- [42] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghahfarokian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, “A survey on deep learning in medical image analysis,” *Medical image analysis*, vol. 42, pp. 60–88, 2017.

-
- [43] K. Gorczowski, M. Styner, J.-Y. Jeong, J. Marron, J. Piven, H. C. Hazlett, S. M. Pizer, and G. Gerig, “Statistical shape analysis of multi-object complexes,” in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2007.
- [44] M. I. Miller, L. Younes, and A. Trouvé, “Diffeomorphometry and geodesic positioning systems for human anatomy,” *Technology*, vol. 2, no. 01, pp. 36–43, 2014.
- [45] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric deep learning: going beyond euclidean data,” *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.
- [46] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [47] B. Knyazev, X. Lin, M. R. Amer, and G. W. Taylor, “Spectral multigraph networks for discovering and fusing relationships in molecules,” *arXiv preprint arXiv:1811.09595*, 2018.
- [48] J. Jung, H.-M. Park, and U. Kang, “Balansing: Fast and scalable generation of realistic signed networks.,” in *EDBT*, pp. 193–204, 2020.
- [49] M. Fey, J. Eric Lenssen, F. Weichert, and H. Müller, “Splinecnn: Fast geometric deep learning with continuous b-spline kernels,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 869–877, 2018.
- [50] T. Kipf, E. Fetaya, K.-C. Wang, M. Welling, and R. Zemel, “Neural relational inference for interacting systems,” *arXiv preprint arXiv:1802.04687*, 2018.
- [51] M. Simonovsky and N. Komodakis, “Graphvae: Towards generation of small graphs using variational autoencoders,” in *International Conference on Artificial Neural Networks*, pp. 412–422, Springer, 2018.
- [52] J. You, R. Ying, X. Ren, W. Hamilton, and J. Leskovec, “Graphrnn: Generating realistic graphs with deep auto-regressive models,” in *International conference on machine learning*, pp. 5708–5717, PMLR, 2018.

-
- [53] R. Liao, Y. Li, Y. Song, S. Wang, C. Nash, W. L. Hamilton, D. Duvenaud, R. Urtasun, and R. S. Zemel, “Efficient graph generation with graph recurrent attention networks,” *arXiv preprint arXiv:1910.00760*, 2019.
- [54] F. R. Chung and F. C. Graham, *Spectral graph theory*, vol. 92. American Mathematical Soc., 1997.
- [55] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, “The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains,” *IEEE signal processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.
- [56] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs,” *arXiv preprint arXiv:1312.6203*, 2013.
- [57] M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” in *Advances in neural information processing systems*, pp. 3844–3852, 2016.
- [58] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.
- [59] Q. Li, Z. Han, and X.-M. Wu, “Deeper insights into graph convolutional networks for semi-supervised learning,” in *Thirty-Second AAAI conference on artificial intelligence*, 2018.
- [60] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” *IEEE transactions on neural networks*, vol. 20, no. 1, pp. 61–80, 2008.
- [61] A. Micheli, “Neural network for graphs: A contextual constructive approach,” *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 498–511, 2009.
- [62] F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein, “Geometric deep learning on graphs and manifolds using mixture model cnns,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5115–5124, 2017.

-
- [63] J. Masci, D. Boscaini, M. Bronstein, and P. Vandergheynst, “Geodesic convolutional neural networks on riemannian manifolds,” in *Proceedings of the IEEE international conference on computer vision workshops*, pp. 37–45, 2015.
- [64] J. Atwood and D. Towsley, “Diffusion-convolutional neural networks,” *Advances in neural information processing systems*, vol. 29, 2016.
- [65] A. Myronenko and X. Song, “Point set registration: Coherent point drift,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.
- [66] X. Ma, S. Xu, J. Zhou, Q. Yang, Y. Yang, K. Yang, and S. H. Ong, “Point set registration with mixture framework and variational inference,” *Pattern Recognition*, vol. 104, p. 107345, 2020.
- [67] N. Duta, A. K. Jain, and M.-P. Dubuisson-Jolly, “Automatic construction of 2d shape models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 5, pp. 433–446, 2001.
- [68] P. J. Besl and N. D. McKay, “Method for registration of 3-d shapes,” in *Sensor fusion IV: control paradigms and data structures*, vol. 1611, pp. 586–606, International Society for Optics and Photonics, 1992.
- [69] J. Zhang, Y. Yao, and B. Deng, “Fast and robust iterative closest point,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [70] W. Lu, G. Wan, Y. Zhou, X. Fu, and S. Song, “Deepicp: An end-to-end deep neural network for 3d point cloud registration. arxiv 2019,” *arXiv preprint arXiv:1905.04153*, 1905.
- [71] R. Zass and A. Shashua, “Probabilistic graph and hypergraph matching,” in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2008.
- [72] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjølness, “New algorithms for 2d and 3d point matching: pose estimation and correspondence,” *Pattern recognition*, vol. 31, no. 8, pp. 1019–1031, 1998.

-
- [73] H. Chui and A. Rangarajan, “A new point matching algorithm for non-rigid registration,” *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, 2003.
- [74] R. Wang, J. Yan, and X. Yang, “Learning combinatorial embedding networks for deep graph matching,” in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3056–3065, 2019.
- [75] A. Zanfir and C. Sminchisescu, “Deep learning of graph matching,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2684–2693, 2018.
- [76] Z. Wang, Q. Lv, X. Lan, and Y. Zhang, “Cross-lingual knowledge graph alignment via graph convolutional networks,” in *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 349–357, 2018.
- [77] Y. Bai, H. Ding, Y. Sun, and W. Wang, “Convolutional set matching for graph similarity,” *arXiv preprint arXiv:1810.10866*, 2018.
- [78] Y. Bai, H. Ding, S. Bian, T. Chen, Y. Sun, and W. Wang, “Simgnn: A neural network approach to fast graph similarity computation,” in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 384–392, 2019.
- [79] T. Heimann and H.-P. Meinzer, “Statistical shape models for 3d medical image segmentation: a review,” *Medical image analysis*, vol. 13, no. 4, pp. 543–563, 2009.
- [80] T. F. Cootes, G. J. Edwards, and C. J. Taylor, “Active appearance models,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 6, pp. 681–685, 2001.
- [81] A. A. Young and A. F. Frangi, “Computational cardiac atlases: from patient to population and back,” *Experimental physiology*, vol. 94, no. 5, pp. 578–596, 2009.
- [82] K.-k. Shen, J. Fripp, F. Mériaudeau, G. Chételat, O. Salvado, P. Bourgeat, A. D. N. Initiative, *et al.*, “Detecting global and local hippocampal shape changes in alzheimer’s disease using statistical shape models,” *Neuroimage*, vol. 59, no. 3, pp. 2155–2166, 2012.

-
- [83] S. Tomoshige, E. Oost, A. Shimizu, H. Watanabe, and S. Nawano, “A conditional statistical shape model with integrated error estimation of the conditions; application to liver segmentation in non-contrast ct images,” *Medical image analysis*, vol. 18, no. 1, pp. 130–143, 2014.
- [84] I. Castro-Mateos, J. M. Pozo, M. Pereañez, K. Lekadir, A. Lazary, and A. F. Frangi, “Statistical interspace models (sims): application to robust 3d spine segmentation,” *IEEE transactions on medical imaging*, vol. 34, no. 8, pp. 1663–1675, 2015.
- [85] F. Cosentino, G. M. Raffa, G. Gentile, V. Agnese, D. Bellavia, M. Pilato, and S. Pasta, “Statistical shape analysis of ascending thoracic aortic aneurysm: correlation between shape and biomechanical descriptors,” *Journal of Personalized Medicine*, vol. 10, no. 2, p. 28, 2020.
- [86] M. Marciniak, A. W. van Deutekom, L. Toemen, A. J. Lewandowski, R. Gailard, A. A. Young, V. W. Jaddoe, and P. Lamata, “A three-dimensional atlas of child’s cardiac anatomy and the unique morphological alterations associated with obesity,” *European Heart Journal-Cardiovascular Imaging*, vol. 23, no. 12, pp. 1645–1653, 2022.
- [87] E. Syrkin, R. Blanc, and G. Székely, “Propagating uncertainties in statistical model based shape prediction,” in *Medical Imaging 2011: Image Processing*, vol. 7962, pp. 1272–1278, SPIE, 2011.
- [88] R. Blanc, C. Seiler, G. Székely, L.-P. Nolte, and M. Reyes, “Statistical model based shape prediction from a combination of direct observations and various surrogates: Application to orthopaedic research,” *Medical image analysis*, vol. 16, no. 6, pp. 1156–1166, 2012.
- [89] G. Harshvardhan, M. K. Gourisaria, M. Pandey, and S. S. Rautaray, “A comprehensive survey and analysis of generative models in machine learning,” *Computer Science Review*, vol. 38, p. 100285, 2020.
- [90] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *Advances in neural information processing systems*, vol. 27, 2014.

-
- [91] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [92] M. Beetz, A. Banerjee, and V. Grau, “Generating subpopulation-specific biventricular anatomy models using conditional point cloud variational autoencoders,” in *International Workshop on Statistical Atlases and Computational Models of the Heart*, pp. 75–83, Springer, 2021.
- [93] P. Romero, M. Lozano, F. Martínez-Gil, D. Serra, R. Sebastián, P. Lamata, and I. García-Fernández, “Clinically-driven virtual patient cohorts generation: An application to aorta,” *Frontiers in Physiology*, p. 1375, 2021.
- [94] M. Danu, C.-I. Nita, A. Vizitiu, C. Suci, and L. M. Itu, “Deep learning based generation of synthetic blood vessel surfaces,” in *2019 23rd International Conference on System Theory, Control and Computing (ICSTCC)*, pp. 662–667, IEEE, 2019.
- [95] B. Gutiérrez-Becker, I. Sarasua, and C. Wachinger, “Discriminative and generative models for anatomical shape analysis on point clouds with deep neural networks,” *Medical Image Analysis*, vol. 67, p. 101852, 2021.
- [96] O. Litany, A. Bronstein, M. Bronstein, and A. Makadia, “Deformable shape completion with graph convolutional autoencoders,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1886–1895, 2018.
- [97] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, “Generating 3d faces using convolutional mesh autoencoders,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [98] J. Yang, K. Mo, Y.-K. Lai, L. J. Guibas, and L. Gao, “Dsg-net: Learning disentangled structure and geometry for 3d shape generation,” *ACM Transactions on Graphics (TOG)*, 2022.
- [99] L. Gao, J. Yang, T. Wu, Y.-J. Yuan, H. Fu, Y.-K. Lai, and H. Zhang, “Sdm-net: Deep generative network for structured deformable mesh,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–15, 2019.

-
- [100] H. Ben-Hamu, H. Maron, I. Kezurer, G. Avineri, and Y. Lipman, “Multi-chart generative surface modeling,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–15, 2018.
- [101] X. Gu, S. J. Gortler, and H. Hoppe, “Geometry images,” in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pp. 355–361, 2002.
- [102] C. Nash, Y. Ganin, S. A. Eslami, and P. Battaglia, “Polygen: An autoregressive generative model of 3d meshes,” in *International Conference on Machine Learning*, pp. 7220–7229, PMLR, 2020.
- [103] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [104] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, Curran Associates, Inc., 2020.
- [105] Z. Liu, Y. Feng, M. J. Black, D. Nowrouzezahrai, L. Paull, and W. Liu, “Meshdiffusion: Score-based generative 3d mesh modeling,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [106] S. Luo and W. Hu, “Diffusion probabilistic models for 3d point cloud generation,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (Los Alamitos, CA, USA), pp. 2836–2844, IEEE Computer Society, jun 2021.
- [107] S. E. Petersen, P. M. Matthews, J. M. Francis, M. D. Robson, F. Zemrak, R. Boubertakh, A. A. Young, S. Hudson, P. Weale, S. Garratt, *et al.*, “UK biobank’s cardiovascular magnetic resonance protocol,” *Journal of cardiovascular magnetic resonance*, vol. 18, no. 1, pp. 1–7, 2015.
- [108] A. Zakeri, A. Hokmabadi, N. Ravikumar, A. F. Frangi, and A. Gooya, “A probabilistic deep motion model for unsupervised cardiac shape anomaly assessment,” *Medical Image Analysis*, vol. 75, p. 102276, 2022.

-
- [109] B. Rister, K. Shivakumar, T. Nobashi, and D. Rubin, “Ct-org: Ct volumes with multiple organ segmentations,” *The Cancer Imaging Archive*, 2019.
- [110] W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3d surface construction algorithm,” *ACM siggraph computer graphics*, vol. 21, no. 4, pp. 163–169, 1987.
- [111] S. Niederer, Y. Aboelkassam, C. D. Cantwell, C. Corrado, S. Coveney, E. M. Cherry, T. Delhaas, F. H. Fenton, A. Panfilov, P. Pathmanathan, *et al.*, “Creation and application of virtual patient cohorts of heart models,” *Philosophical Transactions of the Royal Society A*, vol. 378, no. 2173, p. 20190558, 2020.
- [112] O. Mendeleevitch and M. D. Lesh, “Fidelity and privacy of synthetic medical data,” *arXiv preprint arXiv:2101.08658*, 2021.
- [113] R. H. Davies, C. J. Twining, T. F. Cootes, and C. J. Taylor, “Building 3-d statistical shape models by direct optimization,” *IEEE Transactions on Medical Imaging*, vol. 29, no. 4, pp. 961–981, 2009.
- [114] M. A. Styner, K. T. Rajamani, L.-P. Nolte, G. Zsemlye, G. Székely, C. J. Taylor, and R. H. Davies, “Evaluation of 3d correspondence methods for model building,” in *Biennial International Conference on Information Processing in Medical Imaging*, pp. 63–75, Springer, 2003.
- [115] Z. Niu, G. Zhong, and H. Yu, “A review on the attention mechanism of deep learning,” *Neurocomputing*, vol. 452, pp. 48–62, 2021.
- [116] H. Cai, V. W. Zheng, and K. C.-C. Chang, “A comprehensive survey of graph embedding: Problems, techniques, and applications,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.
- [117] S. Jin, R. R. Lewis, and D. West, “A comparison of algorithms for vertex normal computation,” *The visual computer*, vol. 21, no. 1, pp. 71–82, 2005.
- [118] H. Gouraud, “Continuous shading of curved surfaces,” *IEEE transactions on computers*, vol. 100, no. 6, pp. 623–629, 1971.
- [119] G. Thürrner and C. A. Wüthrich, “Computing vertex normals from polygonal facets,” *Journal of graphics tools*, vol. 3, no. 1, pp. 43–46, 1998.

REFERENCES

- [120] M. Fey and J. E. Lenssen, “Fast graph representation learning with pytorch geometric,” *arXiv preprint arXiv:1903.02428*, 2019.
- [121] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [122] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, “Parametric correspondence and chamfer matching: Two new techniques for image matching,” in *Proceedings: Image Understanding Workshop*, pp. 21–27, Science Applications, Inc, 1977.
- [123] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa, “Laplacian mesh optimization,” in *Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, pp. 381–389, 2006.
- [124] Z. Sabih, S. Khan, *et al.*, “Normal shapes of liver: analysis of normal nuclear scans in multan,” *JPMA. The Journal of the Pakistan Medical Association*, vol. 36, no. 2, pp. 33–36, 1986.
- [125] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*, vol. 4. Springer, 2006.