Shape registration with learned deformations for 3D shape reconstruction from sparse and incomplete point clouds

Xiang Chen [a,b], Nishant Ravikumar [a,b], Yan Xia [a,b], Rahman Attar [a,b], Andres Diaz-Pinto [a,b], Stefan K Piechnik [e], Stefan Neubauer [e], Steffen E Petersen [f,g,h,i], Alejandro F Frangi [a,b,c,d,i,*]

[a] Centre for Computational Imaging and Simulation Technologies in Biomedicine, School of Computing, University of Leeds, Leeds, UK
[b] Biomedical Imaging Department, Leeds Institute for Cardiovascular and Metabolic Medicine, School of Medicine University of Leeds, Leeds, UK
[c] Department of Cardiovascular Sciences, KU Leuven, Leuven, Belgium
[d] Department of Electrical Engineering, KU Leuven, Leuven, Belgium
[e] Oxford Center for Clinical Magnetic Resonance Research (OCMR), Division of Cardiovascular Medicine, University of Oxford, John Radcliffe Hospital, Oxford, UK
[f] William Harvey Research Institute, NIHR Barts Biomedical Research Centre, Queen Mary University London, Charterhouse Square, London, EC1M 6BQ, UK
[g] Barts Heart Centre, St Bartholomew's Hospital, Barts Health NHS Trust, West Smithfield, London, EC1A 7BE, UK
[h] Health Data Research UK, London, UK
[i] Alan Turing Institute, London, UK

ARTICLE INFO

ABSTRACT

Shape reconstruction from sparse point clouds/images is a challenging and relevant task required for a variety of applications in computer vision and medical image analysis (e.g. surgical navigation, cardiac motion analysis, augmented/virtual reality systems). A subset of such methods, viz. 3D shape reconstruction from 2D contours, is especially relevant for computer-aided diagnosis and intervention applications involving meshes derived from multiple 2D image slices, views or projections. We propose a deep learning architecture, coined Mesh Reconstruction Network (MR-Net), which tackles this problem. MR-Net enables accurate 3D mesh reconstruction in real-time despite missing data and with sparse annotations. Using 3D cardiac shape reconstruction from 2D contours defined on short-axis cardiac magnetic resonance image slices as an exemplar, we demonstrate that our approach consistently outperforms state-of-the-art techniques for shape reconstruction from unstructured point clouds. Our approach can reconstruct 3D cardiac meshes to within 2.5-mm point-to-point error, concerning the ground-truth data (the original image spatial resolution is $\sim 1.8 \times 1.8 \times 10\text{mm}^3$). We further evaluate the robustness of the proposed approach to incomplete data, and contours estimated using an automatic segmentation algorithm. MR-Net is generic and could reconstruct shapes of other organs, making it compelling as a tool for various applications in medical image analysis.

1. Introduction

Reconstructing plausible 3D shapes (represented as parametric surface meshes) from sparse, unstructured point clouds (PCs) extracted from single- or multi-view images, is an active problem in Computer Vision (CV) and Medical Image Analysis. 3D shape reconstruction helps visualise the spatial structure of 3D objects, and is relevant to several applications such as, computer-aided diagnosis, surgical planning, image-guided interventions, and com-

putational simulations, to name a few (Suinesiaputra et al., 2017; Lehmann et al., 2009).

Generally, traditional cardiac shape reconstruction comprises two steps: (1) cardiac image segmentation; and (2) mesh generation from the estimated segmentations. Cardiac image segmentation (manual/automatic segmentation) aims to find the region of interest in the original magnetic resonance (MR)/computed tomography (CT) images (e.g. left ventricle (LV), right ventricle (RV)). The mesh generation process then takes the segmentation results as input and generates the corresponding meshes. Marching Cubes (Lorensen and Cline, 1987) is the most widely used algorithm for generating meshes from segmented image volumes, but generally requires dense segmentation volumes for reconstructing

3D shapes as triangulated surfaces/meshes. Such an approach is ill-suited to reconstructing 3D shapes from sparse, stacked 2D contours. Therefore, in cardiac shape reconstruction, previous studies have approached the problem as one of mesh adaptation. In this context, a template mesh is first generated (either using an isosurfacing technique or directly from an existing statistical atlas/template), and then deformed under the guidance of contours or points (extracted from segmentation results) (Medrano-Gracia et al., 2013; Lim et al., 2014; Zou et al., 2015; Villard et al., 2018). Using segmented contours to deform the template mesh, Lim et al. (2014) proposed to reconstruct specific 4D meshes (spatial-temporal mesh) for patients. Villard et al. (2018) proposed a method to reconstruct geometrical surface meshes from sparse, heterogeneous, non-coincidental contours. They used contours to guide the deformation of an initial mesh to obtain the target mesh, using a smoothness term while maximising the data fitting. However, those methods are all time-consuming, which limits mesh reconstruction for real-time applications in surgical guidance and navigation.

Deep learning-based methods have also been explored for this task. As inference using a trained deep neural network is just one forward pass through the network, such methods can significantly speed up the process of cardiac shape reconstruction. Few studies have explored the application of deep learning methods on this task. For example, Xu et al. (2019) proposed to tackle this task as a volumetric mapping problem followed by isosurface estimation using the generated volume. Their approach generated three dense 3D volumes, LV myocardium, LV cavity and RV cavity, from sparse volumes of contours. Then marching cubes was used to reconstruct the bi-ventricular cardiac meshes. This approach was able to accurately predict cardiac meshes even with discrepancies between intersecting slices (short-axis (SAX) view and long-axis (LAX) view slices). Attar et al. (2019) viewed shape reconstruction as a regression problem, building a deep regression network to predict the cardiac shape parameters in Principal Component Analysis (PCA) space from image data (from the UK Biobank (UKBB) cohort), using both short and long axis views and patient metadata. Using a cardiac statistical shape model (SSM) estimated *a priori* and its associated mean template mesh and principal eigenvectors, during inference, they reconstructed the bi-ventricular cardiac meshes for each unseen image volume using the PCA parameters estimated by their network. Instead of using traditional methods to generate the final shape, some studies have proposed to predict cardiac PCs (Zhou et al., 2019) or meshes (Wang et al., 2020) directly using deep neural networks, enabling cardiac shape reconstruction in real-time. Zhou et al. (2019) firstly proposed to apply deep learning network in cardiac point cloud reconstruction, which could reconstruct RV from a single image (in the LAX view). Similarly, Wang et al. (2020) designed a deep learning network, Instantiation-Net, to reconstruct 3D RV mesh based on a single LAX view image. However, reconstructing a 3D object from the image in a single view is ill-posed due to the large proportions of missing information, making it difficult to generate accurate meshes.

To reconstruct plausible and high-quality meshes from cardiac images, multiple images with boundary information (e.g. contours) would be a better input choice. They are usually available from manual/semi-automatic contours derived from most medical image segmentation tools. As those tools do not provide full 3D reconstructions, the mesh reconstruction method could be a supplement of these tools in return. Previous research (Medrano-Gracia et al., 2013; Lim et al., 2014; Zou et al., 2015; Villard et al., 2018) have also proved that deforming a template mesh under the guidance of contours facilitates the generation of high-quality personalised meshes (fitted to the contours). Therefore, in this paper, we focus on cardiac mesh reconstruction from a point cloud of contours. We design a novel approach, MR-Net, to achieve the task

of reconstructing 3D bi-ventricular cardiac shapes from stacked 2D contours, viewing it as a DL-based template-to-PC fitting task. An overview of the proposed framework is presented in Fig. 1. Given SAX cine-cardiac MR image stacks, we first manually/automatically segment the cardiac structures of interest in each 2D slice. Next, PCs of stacked contours are extracted from these segmentations. Finally, MR-Net is applied to predict high-quality meshes from PCs of contours. With deep learning-based segmentation methods and MR-Net, we can reconstruct accurate 3D cardiac shapes from the MR images accurately, robustly, and in real-time.

Recently, many deep learning-based methods have been proposed for meshes/PCs reconstruction and analysis (Qi et al., 2017a; 2017b; Wang et al., 2018). Among them, the most popular task is to reconstruct 3D mesh from single-/multi-view image(s). Wang et al. (2018) firstly proposed a network Pixel2mesh based on graph convolutional network (GCN) (Bronstein et al., 2017; Defferrard et al., 2016; Kipf and Welling, 2017) for mesh reconstruction from a 2D image (a projection of the original 3D object on to one view). They used an ellipsoid mesh as the template, then applied the GCN blocks to deform it with the guidance of features extracted from the input image using VGG 16-like architecture (Simonyan and Zisserman, 2015) like architecture. Based on it, Wen et al. (2019) proposed an improved network, Pixel2mesh++, to tackle the problem of 3D mesh reconstruction from multi-view images, reconstructing more accurate surfaces of 3D objects. Instead of GCN, Pan et al. (2019) proposed to apply a multi-layer perceptron (MLP) as the deformation module followed by topology modification blocks, and finally designed a boundary refinement block to improve the visual quality of reconstructed meshes further. These approaches were developed and validated on publicly available datasets for the reconstruction of general objects (e.g. plane, chair). Using approaches like Pixel2mesh, recent studies have also explored the reconstruction of human hand (Ge et al., 2019) or body (Kolotouros et al., 2019) meshes from 2D images. In addition, several deep learning-based methods have been proposed for mesh reconstruction from dense PCs, which rely on predicting the surface normal vector for every point in the input PCs (Hashimoto and Saito, 2019), or predicting the skinned multi-person linear model (SMPL, i.e. a parametric human body model (Loper et al., 2015)) parameters of the target mesh, then using the off-the-shelf SMPL model to reconstruct meshes from parameters (Jiang et al., 2019). However, in our case, the input PCs are sparse contours with large proportions of missing information relative to dense point cloud-based representations of shapes. And these contour points differ in number and spatial distribution to the vertices of the surface (our target/output) that they implicitly represent.

To this end, considering the nature of the traditional cardiac mesh reconstruction methods and the context of deep learning-based mesh reconstruction methods, we propose to use deep learning network to deform a cardiac template mesh to obtain the target meshes under the guidance of contours. The key idea behind mesh reconstruction from single/multiple images is to find a mapping from the input image(s) to the template mesh, and subsequently, to use the learned features in the input image to guide the deformation of the template mesh. Generally, a 2D projective transformation is applied to find the corresponding pixels in the 2D image for every vertex in the template mesh, before transferring the features of 2D pixels to the corresponding vertex. However, in our case, the inputs are contour PCs in 3D coordinate space. Applying a single 2D projection of the input PCs would cause a loss of structural information. Therefore, we design a PC-to-PC mapping going from the 3D contour point cloud to a 3D volume, and correspondingly, from the 3D volume to the vertices of the 3D template mesh (i.e. a PC-volume-PC mapping), which addresses the challenge of mapping features between unstructured data sets that lack spatial correspondence.
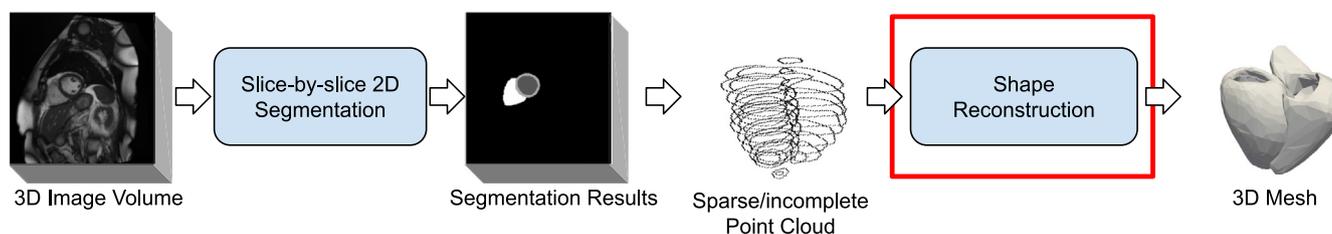
Fig. 1. The proposed pipeline for 3D cardiac shape reconstruction from MR images (The cardiac image presented were reproduced with the permission of UK Biobank©). Note that, the slice-by-slice segmentation methods can be both manual segmentation and automatic segmentation algorithms.

The main contribution of our work is a hybrid graph convolutional neural network for 3D mesh reconstruction, MR-Net, which approaches the problem as a template deformation task conditioned on the sparse point cloud data (stacked 2D contours in our case). To the best of our knowledge, this is the first study to employ deep learning for registering a 3D mesh to sparse PCs (or stacked 2D contours), enabling real-time 3D shape reconstruction. Although we focus on 3D cardiac shape reconstruction from stacked 2D contours in this study, MR-Net is generic and flexible, and can be employed for various PC-to-PC/mesh reconstruction tasks (e.g. PC/mesh reconstruction, PC/mesh completion and correction) within the medical imaging or CV domain. To sum up, the contributions of this paper are as follows,

- We propose a novel cardiac mesh reconstruction framework, which can predict accurate cardiac meshes from original MR images in a fast and robust manner, assisted by existing deep learning-based segmentation methods.
- We demonstrate that MR-Net can generate accurate and high-quality meshes even from incomplete contours, a challenge that often arises in clinical scenarios.

The rest of the paper is organised as follows: In Section 2, each component of the proposed approach is described. Section 3 exemplifies our proposed MR-Net on UKBB dataset. Finally, Section 4 is the conclusion of this paper.

2. Methods

Traditional 3D shape reconstruction approaches have relied on iterative deformation of a template mesh to sparse contours/PC, using the latter to guide the former, with including various penalty terms to ensure the estimated deformation is smooth. To eliminate the requirement of several iterations during inference (which can be time-consuming), in this paper, a deep learning-based network, MR-Net, is designed to mimic such a process. After training, unseen contours/PCs are reconstructed into 3D shapes (represented as triangulated surface meshes) via a simple forward pass through the network. This can significantly speed up 3D shape reconstruction while predicting high-quality meshes. In subsequent sub-sections, we first introduce the overall network architecture of MR-Net, and then provide details of — the feature extraction module, deformation module, 3D PC-to-PC mapping, and the loss function formulated for effective training of the proposed approach.

2.1. Network architecture

The task of our MR-Net is to reconstruct personalised meshes from sparse contours under the guidance of a template mesh. To accomplish this, we design two modules: the feature extraction module and the deformation module (comprising three GCN blocks), as shown in Fig. 2. The purpose of the feature extraction module is to extract features from the input point cloud of stacked contours that are beneficial for the deformation module, while the latter utilises this information to deform the template mesh to the

personalised target mesh under the guidance of the features from the feature extraction module. The feature extraction module consists again of two parts: direct PC feature extraction (in PC domain), and 3D convolutional neural network (CNN LeCun et al., 1998) feature extraction (in image domain). The former is to extract features directly from point clouds whereas the latter extracts features from a voxel-based representation of the contours.

Generally, the meshes can be presented by vertices and connectivity. Following Pixel2mesh (Wang et al., 2018), we assume the connectivity in the target meshes are fixed (the same as template mesh), thereby the mesh reconstruction from PC could be simplified to learn the mapping between input PC and vertices of target meshes. To achieve this mapping, two problems must be addressed: (1) how to learn the shape priors from input PC; (2) how to find the point-to-point correspondence between the input PC and vertices of the template, in order to apply the graph convolution. The main contributions of our proposed MR-Net lie to tackle these two challenges.

2.2. Feature extraction

Due to the large proportion of missing inter-slice information, 3D shape reconstruction from sparse 2D contours is a challenging task. A template mesh is randomly selected from the training dataset to supply the missing information in the reconstruction process. The input PCs of contours serve as the guidance of template deformation. All the input PCs and corresponding target meshes are normalised to a standard sphere (centred at $(0,0,0)$ with the radius of 1) before training the network. To learn the guidance information, feature extraction from the input contours is decomposed into two paths.

The first path is a point cloud feature extraction block based on PointNet++ Qi et al., 2017b), which predicts two new PCs using sampling and grouping. In our experiments, the number of points in input PCs is 3,000, and these two new PCs contain 2,000 and 1,578 points respectively (the number of points is set empirically, sampling and grouping the original point clouds of contours gradually from 3,000 to 1,578). After obtaining these two new PCs, a 3D projection (i.e. a mapping from vertices' coordinate to index of voxels, projecting points in 3D space to voxels in 3D volume, see in Formula. (2) is applied to transfer them with the original point cloud of contours into three 64^3 features, where each voxel is a feature vector with dimension 1×4.

In the other path, we first apply a 3D projection to turn the unstructured input point cloud into a structured volume with 64^3 voxels in the image domain. Then a 3D CNN (4 layers, downsampling from 64^3 to 8^3) is used to extract features from the 3D volume projected from the input point cloud, where the extracted features contain feature maps in all four resolutions $(64^3, 32^3, 16^3, 8^3)$, where corresponding feature dimensions are 64, 128, 256, 500 respectively.

With a volume-to-PC mapping, we can map the features in voxels of volumes back to points in the template mesh and guide its deformation. Therefore, we finally obtain a feature of $(64 + 128 +$
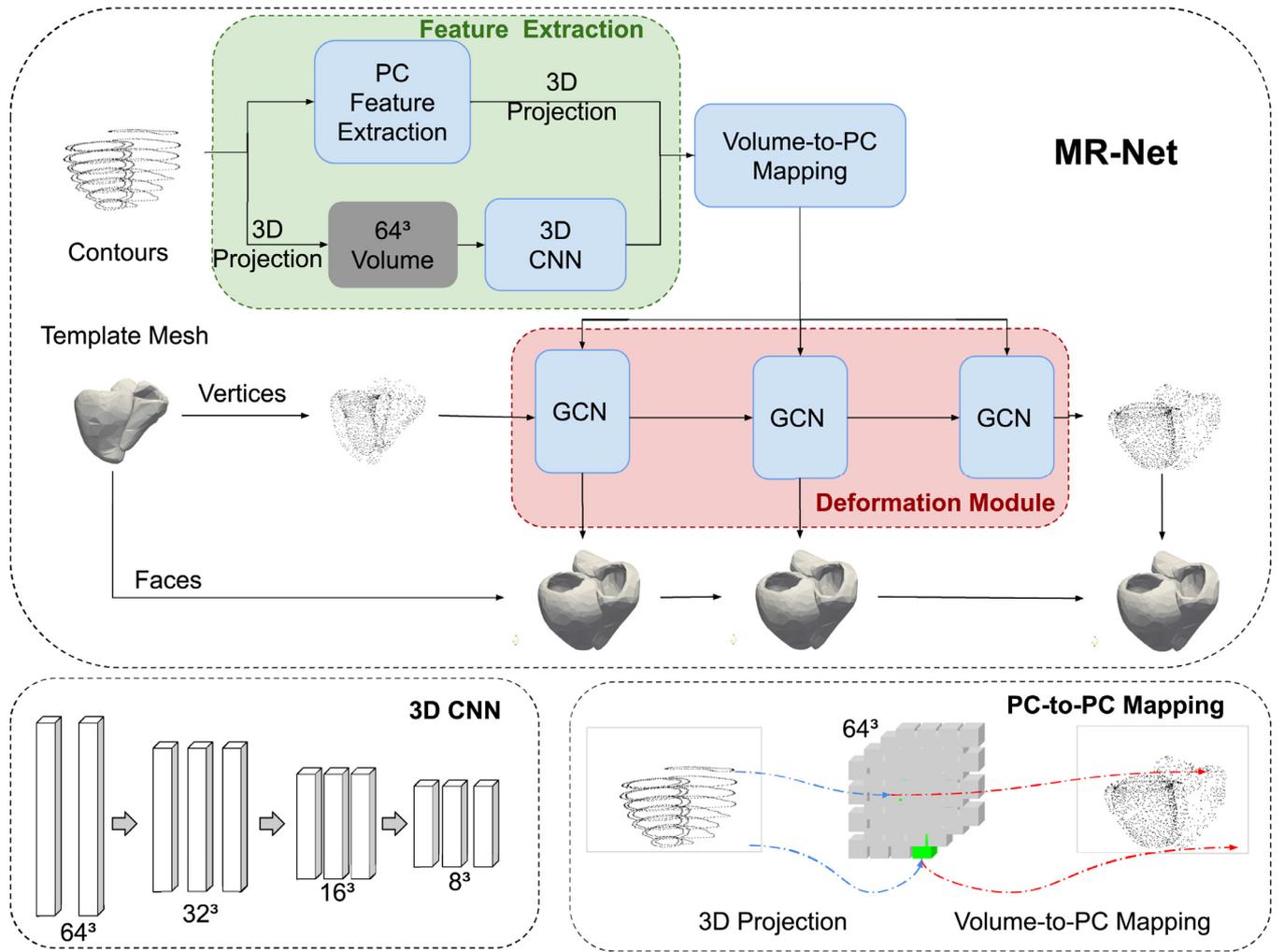
Fig. 2. Schema of the proposed method, MR-Net. The overall network is displayed in the top row, with the details of the 3D CNN and our proposed PC-to-PC mapping blocks (between input PCs and vertices of template mesh) presented in the bottom row. The feature extraction module is to extract features from the input contours, and then the deformation module deforms the template mesh to the target mesh under the guidance of the learned features in features extraction module.

$256 + 500 + 4 \times 3) = 960 \times 1$ for every point in the vertices of template mesh, which is concatenated with the coordinate (3×1) of the template mesh (or the coordinate predicted in the previous GCN block) and taken as input by the GCN blocks. Although it would cause little information missing in the process of 3D projection, the multi-layer 3D CNN learns rich structured features (across different resolutions) from the original PCs, which is essential for extracting features from the input PCs. With feature extraction in both point cloud domain and image domain, we can obtain a proper understanding of the input contours and use it to guide the deformation of the template mesh.

2.3. Deformation module

With the features learned from input contours as the guidance, we design a deformation module to deform the template mesh gradually, which helps to preserve the topology and the connectivity of meshes, following deformation. The deformation module includes three GCN blocks (referring to Pixel2mesh (Wang et al., 2018)), each comprising 14–15 graph convolution layers (the first is 14, while the next two are 15). Note that, the number of layers in MR-Net are set empirically and tuned based on results obtained on the validation set.

3D meshes comprise vertices, edges and faces. The vertices are the coordinates of the nodes on the mesh, which is generally an $N \times 3$ array (the three columns stand for x,y,z coordinate respectively). Edge denotes the connectivity between two vertices. In our case, the face of the mesh is defined by surface triangles, whereby, every face in the mesh comprises the indices of three vertices (connected by edges to form a triangle). Let $\mathbf{F} = \{\mathbf{f}_i\}_i^N$ be the features on every vertex of the mesh, the graph convolution layer can be formulated as,

$$\mathbf{f}_{\mathbf{p}}^{l+1} = \omega_0 \mathbf{f}_{\mathbf{p}}^l + \sum_{\mathbf{q} \in N(\mathbf{p})} \omega_1 \mathbf{f}_{\mathbf{q}}^l, \tag{1}$$

where $\mathbf{f}_{\mathbf{p}}^{l+1} \in \mathbb{R}^{d_{l+1}}$ is the output feature of vertex $\mathbf{p}$ after l-th graph convolution layer, and $\mathbf{f}_{\mathbf{p}}^l \in \mathbb{R}^{d_l}$ is the corresponding input feature in l-th layer. $N(\mathbf{p})$ are the neighbour points of vertex $\mathbf{p}$. Both ω_0 and ω_1 are parameters ($d_l \times d_{l+1}$) automatically learned during training. The ω_1 is shared by all edges, thereby the graph convolution layer can be applied to meshes with irregular shapes (i.e. nodes with different vertex degrees).

The structure of GCN blocks mainly follows Pixel2mesh (Wang et al., 2018). In the first GCN block, the first graph convolution layer takes the concatenation of the learned feature (1×960) and the original vertices (1×3) of template mesh as input and predicts hidden features at 1×256, followed by 12 hidden graph convolution layers (the input is 1×256 and the output is 1×256) and a graph convolution layer to predict the coordinate of each

vertex (1×3). The next two GCN blocks are the same, where the first graph convolution layer takes the concatenation of learned contour features (1×960), the predicted coordinates (1×3) and the learned features (1×256) in hidden layers of the previous GCN block as input and predicts features at 1×256. This is followed by 13 hidden graph convolution layers (both input and output are 1×256) and a graph convolution layer to predict the coordinates (1×3). Therefore, each GCN block predicts an output of the target mesh, while the template mesh is deformed gradually to fit the contours. Further details about GCN can be found in Wang et al. (2018); Bronstein et al. (2017).

2.4. 3D PC-to-PC mapping

To apply deformation based on GCN, point-level features are required for the vertices in the template mesh. However, as the input point cloud and the template mesh are both unstructured and have different cardinalities, there is no point-to-point correspondence between them. To transfer the learned shape information from input point cloud to the vertices of the template mesh, we build a PC-to-PC mapping module comprising 3D projection and volume-to-PC mapping, where the 3D volume is used as the bridge between the input point cloud and template. The 3D projection aims to map 3D PCs to 3D volumes, which can be formulated as follows (using a volume of 64^3 voxels as an example),

$$V_{x,y,z} = \begin{cases} 0, & (x,y,z) \neq \lfloor (\mathbf{P}_i) \times 32 \rfloor + 32 \\ 1, & (x,y,z) = \lfloor (\mathbf{P}_i) \times 32 \rfloor + 32 \end{cases} \tag{2}$$

where $\mathbf{P}_i$ is the coordinate of i-th point in PCs, which has been normalised before the training. $V_{x,y,z}$ is the corresponding voxel in projected 3D volumes. We project the point cloud into a 64^3 volume. If there is a corresponding point in the point cloud, the voxel in 3D volume would be 1, otherwise 0.

Correspondingly, the volume-to-PC mapping is the inverse process of 3D projection,

$$\mathbf{f}_i = \mathbf{VF}_{x,y,z}, s.t.(x,y,z) = \lfloor (\mathbf{P}_i) \times 32 \rfloor + 32, \tag{3}$$

where $\mathbf{f}_i$ is the obtained feature for point i in template mesh, and $\mathbf{VF}_{x,y,z}$ is the corresponding feature in 3D volume. With these two mappings, we finally obtain point-level features for the template mesh, which serve as the input of GCN blocks. Note that, there is a coordinate scale missing (from float coordinate to integer index) in the process of 3D PC-to-PC mapping. Generally, larger volumes would enable more accurate reconstruction results, although requiring more memory. For a trade-off between the accuracy and computational complexity (GPU memory), we choose a 64^3 volume as the bridge between input PC and template mesh.

2.5. Loss functions

We employ deep supervision with a multi-term mesh loss function to train our proposed MR-Net. The mesh loss is designed following Pixel2mesh (Wang et al., 2018), including Chamfer distance (CD), edge loss, normal loss and Laplacian loss. CD is applied to capture an overall distance between the predicted vertices and vertices of ground-truth. It does not require the point number/order to be the same in the two PCs. Denoting $\mathbf{p}$ and $\mathbf{q}$ as the predicted and ground-truth vertices, Chamfer distance L_{CD} is written as,

$$L_{CD} = \sum_{\mathbf{p}} min_{\mathbf{q}} ||\mathbf{p} - \mathbf{q}||_2^2 + \sum_{\mathbf{q}} min_{\mathbf{p}} ||\mathbf{p} - \mathbf{q}||_2^2. \tag{4}$$

Edge loss is a regularisation to penalise high edge length. We use the sum of all edge lengths in the predicted mesh as the edge loss L_{edge},

$$L_{edge} = \sum_{\mathbf{p}} \sum_{\mathbf{k} \in N(\mathbf{q})} ||\mathbf{p} - \mathbf{k}||_2^2, \tag{5}$$

where $N(\mathbf{q})$ is the neighbour vertices of $\mathbf{q}$.

Normal loss L_{normal} is computed on surface normals, which helps preserve mesh topology and retain fine structural details, and is formulated as,

$$L_{normal} = \sum_{\mathbf{p}} \sum_{\mathbf{q}=argmin_{\mathbf{q}}(||\mathbf{p}-\mathbf{q}||_2^2)} || < \mathbf{p} - \mathbf{k}, \mathbf{n_q} > ||_2^2, s.t. \mathbf{k} \in N(\mathbf{p}), \tag{6}$$

where $< \cdot, \cdot >$ is the inner product of two vectors, $\mathbf{k}$ belongs to the neighbour point of $\mathbf{p}$ (denoted by $N(\mathbf{p})$), and $\mathbf{n}_q$ is the surface normal of ground-truth. In the predicted/target meshes, the vectors (edges) from each vertex to its neighbour vertices should be perpendicular to its normal. If the predicted vertices of meshes are exactly the same as the target mesh, the normal loss becomes zero. Therefore, this loss is to guarantee the normal of the predicted mesh is close to the normal in the target mesh.

Similar to edge loss, Laplacian loss $L_{Laplacian}$ is also a regularisation term. Let $\delta_{\mathbf{p}}$ be the Laplacian coordinate of vertex $\mathbf{p}$. The $L_{Laplacian}$ is as follows,

$$\delta_{\mathbf{p}} = \mathbf{p} - \sum_{k \in N(\mathbf{p})} \frac{1}{||N(p)||} \mathbf{k},$$
$$L_{Laplacian} = \sum_{\mathbf{p}} ||\delta_{\mathbf{p}}' - \delta_{\mathbf{p}}||_2^2, \tag{7}$$

where $\delta_{\mathbf{p}}$ and $\delta_{\mathbf{p}}'$ are the Laplacian coordinates of vertex $\mathbf{p}$ before and after deformation.

The mesh loss has been proven to be useful in mesh reconstruction (Wang et al., 2018; Wen et al., 2019). However, in our task, we found it is inadequate to generate accurate vertex coordinates, as there is no exact point-to-point loss. To tackle this issue, we further apply an additional $L1$ loss between the predicted and ground-truth vertices. This term ($L1$) urges MR-Net to predict more accurate vertices for the reconstructed cardiac mesh, and it is formulated:

$$L1 = \frac{1}{M} \sum_i^M |\mathbf{p}_i - \mathbf{q}_i|, \tag{8}$$

where M is the number of points in the predicted mesh. $\mathbf{p}_i$ and $\mathbf{q}_i$ are coordinates of the i-th point in the predicted and target mesh, respectively.

Therefore, the complete mesh loss L_{mesh} we propose is as follows,

$$L_{mesh} = L_{CD} + L_{edge} + L_{norm} + L_{Laplacian} + \lambda_0 \times L1, \tag{9}$$

where λ_0 is a hyper-parameter that needs to be tuned empirically.

As there are three outputs in MR-Net from coarse to fine, we compute the mesh loss on all three outputs. Therefore, the final loss function L_{total} is computed as follows,

$$L_{total} = \lambda_1 L_{mesh1} + \lambda_2 L_{mesh2} + \lambda_3 L_{mesh3}, \tag{10}$$

In the loss function, λ_0, λ_1, λ_2 and λ_3 are hyper-parameters that weight the relative influence of each structural loss term, on the overall gradient backpropagated through the network to update the constituent weights. These weights are also tuned empirically.

3. Experiments and results

3.1. Data and implementation

All experiments conducted to validate MR-Net are performed using 7,870 stacks of 2D contours, available from the manual delineation of SAX view cardiac MR images (at the end of systole and diastole), within the UKBB dataset. The spacing for cardiac MR images in UKBB is 1.8×1.8 mm^2 with a slice thickness of 8.0 mm

and a slice gap of 2 mm. Manual contouring was performed by a team of cardiac imaging experts (Petersen et al., 2017) and the corresponding 3D bi-ventricular cardiac reference shapes were available from a previous study (Attar et al., 2019). We randomly split the dataset into training (6,000), validation (935) and test sets (935). Each training sample comprises a source-target pair, where the former is the sparse 2D contour points to be reconstructed, while the latter is the corresponding bi-ventricular surface mesh (i.e. the target shape). We pre-process all source PCs to maintain the same cardinality (3,000 points used in all experiments) across all samples. This is done by replicating points at random for samples with less than 3,000 points. The target mesh vertices however, all have the same cardinality (i.e. 1,578 points) and consequently need not be resampled. In the training dataset, all input PCs and target mesh vertices are normalised before training the network, using their centroid and radius (fixed as 100.00 mm). Therefore, all the PCs and meshes used for training are normalised to a sphere centred at (0,0,0) with a radius of 1. Correspondingly, during testing, the input PCs are also normalised before shape reconstruction, such that the predicted meshes can be transformed to their original size using the same values for the centroid and radius.

We use the Adam optimiser, with a learning rate of $1e^{-05}$ and a batch size of 1 to train MR-Net, in all experiments conducted. The hyper-parameters λ_0, λ_1, λ_2 and λ_3 for the total structural loss are 1,000, 0.1, 0.3 and 0.6 respectively, which are determined empirically. Note that, these parameters are the same in all experiments. The network is implemented using Python and TensorFlow, and all experiments are streamlined and executed on Tesla M60 GPUs, accessed over the MULTI-X platform[1] (de Vila et al., 2018). All networks are trained until convergence on the training set. It takes $\sim$ 80h to train MR-Net on data with complete contours, and $\sim$ 100h on training data of incomplete contours (as the training number is larger). Our source code will be available on the Github[2] following acceptance of this paper.

3.2. Comparison with the state-of-the-art

To the best of our knowledge, no previous deep learning-based method for mesh reconstruction from (stacked) contour PCs exists in literature. However, various techniques such as point cloud up-sampling, point cloud segmentation and mesh reconstruction from a single image could be modified to partially address the reconstruction problem. Consequently, we build three baselines, using state-of-the-art networks for comparison, namely, PontNet++ (Qi et al., 2017b), PU-Net (Yu et al., 2018), and Pixel2mesh (Wang et al., 2018). PointNet++ is a popular network originally proposed for point cloud classification and segmentation. We build an encoder-decoder network based on its kernel block, with a feature integration component, to obtain our baseline network PointNet++. PU-Net is a state-of-the-art network for point cloud up-sampling. We adapt it to point cloud reconstruction by incorporating a sampling layer at the end of the original network. These two networks can predict PCs only with similar structures to the ground-truth, but cannot recover the cardiac mesh as the order predicted points differ from the ground-truth. To compare our approach with mesh reconstruction methods, we project the input PCs onto 2D images, and then reconstruct 3D cardiac meshes from them using Pixel2mesh (Wang et al., 2018). In addition to deep learning-based methods, we also compare our MR-Net with two traditional point set registration methods, coherent point drift (CPD Myronenko and Song, 2010) and GMMREG (Jian and Vemuri, 2011), where the template mesh is the same as MR-Net and

the hyper-parameters are tuned based on samples from the training and validation set.

3.2.1. Qualitative results

A visual comparison of the generated reconstructions using the proposed method against the baseline networks is depicted in Fig. 3. For Pixel2mesh, CPD and our proposed MR-Net, both predicted meshes and the corresponding vertices (PCs) are presented, while, only PCs are available for PointNet++ and PU-Net results. For GMMREG, only the mesh is presented, due to limited space. As shown in Fig. 3, the PointNet++ and PU-Net reconstructions still contain several "contour-like" distributions of points and lack the inlet to the pulmonary artery at the top of the RV. The reconstruction of Pixel2mesh just learns a coarse representation of the cardiac shape, and the corresponding mesh does not preserve bi-ventricular topology and is thus significantly different from the ground-truth. The main reason for this is that 2D projection causes a significant loss in information, resulting in erroneous reconstructions. It is difficult for Pixel2mesh to reconstruct meshes with holes using 2D information only. Both traditional point set registration methods, CPD and GMMREG can reconstruct smooth cardiac meshes, whilst preserving topology. In our task, the performance of CPD is better than GMMREG. However, the mesh obtained using CPD is significantly different to the ground-truth mesh, failing to capture several local details (mainly on the top and bottom of the ventricles). MR-Net can reconstruct evenly distributed PCs without contour-like artefacts, while preserving bi-ventricular topology and retaining fine structural details such as the inlet to the pulmonary artery. The reconstructed mesh is of high-quality and more closely matches the target shape, compared with the other approaches. This is further supported by the quantitative results summarised in the next section.

3.2.2. Quantitative results

The reconstruction performance of MR-Net is also quantitatively evaluated and compared with other baseline networks. Following previous shape reconstruction research (Zhou et al., 2019; Li et al., 2019; Wang et al., 2018), reconstruction accuracy was measured using CD, earth mover's distance (EMD), Hausdorff distance (HD) (Li et al., 2019; Yu et al., 2018; Wang et al., 2018) and point cloud to point cloud (PC-to-PC) error (Zhou et al., 2019; Wang et al., 2020), which could capture the distance between two PCs from different perspectives. The CD, EMD, and HD are well-known metrics to evaluate the distance between two PCs, while PC-to-PC error is computed as,

$$\epsilon_{PC-to-PC} = \frac{1}{M} \sum_{m=1}^{M} \sqrt{\sum_{i=1}^{3} (\mathbf{p}_{m,i} - \mathbf{q}_{m,i})^2}. \tag{11}$$

where $\mathbf{p}$ and $\mathbf{q}$ are vertices of predicted meshes and ground-truth, and the M is the number of points in predicted meshes (in our experiments is 1,578). For all evaluation metrics, lower values signify better performance. The average reconstruction accuracy (expressed as mean±std) across all test samples is summarised in Table 1, for each approach, as mentioned earlier. Paired sample t-tests were used to assess statistical significance, by comparing the reconstruction accuracy of each baseline network with that of MR-Net. MR-Net consistently outperformed the others, achieving the best results across all metrics. Note that, CPD is also the method used to generate the ground-truth meshes (as mentioned in Attar et al. (2019)), requiring a time-consuming process of tuning hyper-parameters for each sample. In this paper, for comparison, we tune the hyper-parameters based on several samples from training and validation set, and use the same hyper-parameters for all testing samples. This is why the meshes obtained using CPD in

---
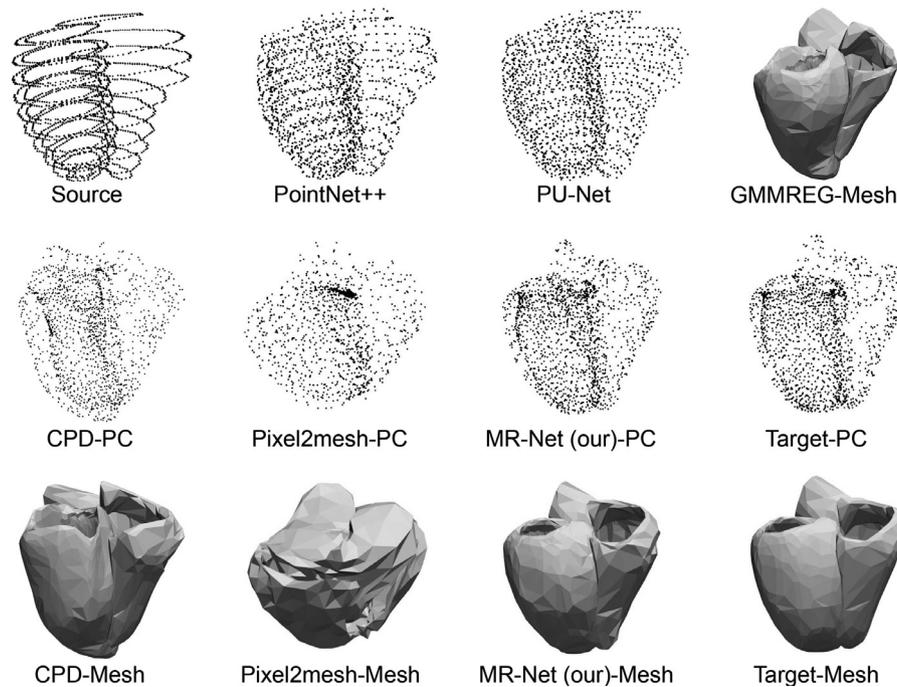
Fig. 3. Qualitative results for our MR-Net and baseline networks viz. PointNet++, PU-Net, GMMREG, CPD and Pixel2mesh. In the second and third row, PCs and meshes computed using MR-Net, CPD and Pixel2mesh are presented.

Table 1
Quantitative comparison between MR-Net and the baseline networks using the CD, EMD, HD and PC-to-PC error. Statistically significant differences in reconstruction accuracy are highlighted in bold. MR-Net (automatic) represents the mesh reconstruction results from contours extracted using automatic segmentation methods (see Section 3.4).

Methods	CD (mm)	EMD (mm)	HD (mm)	$\epsilon_{PC-to-PC}$ (mm)	Inference Time(s)
PointNet+	13.03 ± 2.96	17.94 ± 2.07	17.04 ± 3.57	-	< 0.1
PU-Net	12.15 ± 2.88	14.94 ± 2.02	15.74 ± 3.37	-	< 0.1
Pixel2mesh	19.38 ± 5.54	25.27 ± 4.48	16.20 ± 3.30	50.63 ± 7.29	< 0.1
CPD	12.10 ± 6.63	12.49 ± 5.46	13.05 ± 7.74	7.03 ± 2.94	37.45
GMMREG	20.90 ± 7.18	17.58 ± 4.85	15.87 ± 3.04	8.36 ± 1.85	60.90
MR-Net (No L1)	255.08 ± 94.54	36.61 ± 5.49	47.80 ± 7.35	39.12 ± 5.21	< 0.1
MR-Net (Only L1)	6.14 ± 1.61	7.01 ± 1.48	8.10 ± 1.79	3.34 ± 0.65	< 0.1
MR-Net (No PC feature)	6.84 ± 1.69	8.07 ± 1.64	8.78 ± 1.92	3.87 ± 0.65	< 0.1
MR-Net (No 3D CNN)	80.71 ± 39.28	32.03 ± 7.27	29.63 ± 7.06	18.54 ± 2.68	< 0.1
MR-Net	$\mathbf{4.39 \pm 1.48}$	$\mathbf{5.05 \pm 1.41}$	$\mathbf{6.89 \pm 1.88}$	$\mathbf{2.48 \pm 0.63}$	< 0.1
MR-Net (automatic)	7.57 ± 3.59	8.19 ± 2.87	9.31 ± 2.86	3.45 ± 0.98	< 0.1
MR-Net (small dataset)	6.89 ± 1.76	8.12 ± 1.71	8.83 ± 2.00	3.92 ± 0.79	< 0.1

this study are different to the target meshes (generated in a previous study (Attar et al., 2019)). As the inference of MR-Net is much faster (< 0.1 s vs 37.45/60.90 s) and more accurate than traditional methods, there is potential for its use in real-time applications.

To further demonstrate the clinical potential and superiority of our approach, we extract the corresponding segmentations from the predicted and ground-truth meshes (using the SAX-planes from the original cardiac MR images), and compute five clinical indices based on the obtained segmentation results - LV end-diastolic volume (LVEDV), end-systolic volume (LVESV), LV stroke volume (LVSV), LV ejection fraction (LVEF) and LV myocardial mass (LVM) respectively. The clinical indices are shown in Table 2 (as topology is not preserved in meshes predicted by pixel2mesh and MR-Net(No L1), we did not include their clinical indices), where, values showing no statistically significant difference to the clinical indices computed on the ground-truth meshes are highlighted in bold ($p \geq 0.05$). While the meshes predicted by MR-Net incur an average point-to-point error of $2.48mm$ to the ground-truth, we found that the computed clinical indices for MR-Net show no significant difference to the latter. All other approaches investigated on the other hand, show significant differences to the ground truth, in

terms of the clinical indices evaluated. This further demonstrates the superiority of our approach at preserving key clinical indices that are routinely used to assess cardiac function.

We also explore the performance of MR-Net when trained with a limited number of samples, as 6,000 samples are not easy to obtain in real clinical applications. We randomly choose 200 samples from the original training set to train MR-Net and evaluate its performance with the same test set. The results are shown in Table 1(MR-Net(small dataset)). These results indicate that MR-Net performs well in the small data regime, and outperforms other state-of-the-art methods which were trained on a significantly larger sample size (6,000).

These quantitative results follow the visual assessment (cf. Section 3.2.1) of the bi-ventricular shapes reconstructed using each approach. This highlights further the efficacy of our proposed MR-Net for 3D shape reconstruction from stacked 2D contours.

3.3. Shape reconstruction from incomplete contours

Typical artefacts encountered during cardiac MR image acquisition include missing slices between the base and apical of the

Table 2

Clinical indices (LVEDV, LVESV, LVSV, LVEF, LVM) computed based on the segmentation obtained from the predicted meshes. Those clinical indices make no statistically significant difference to the ground-truth (GT) are highlighted in bold ($p \geq 0.05$).

Methods	LVEDV (ml)	LVESV (ml)	LVSV (ml)	LVEF(%)	LVM(g)
CPD	77.66 ± 17.57	43.79 ± 11.86	33.87 ± 9.05	43.71 ± 7.09	161.71 ± 36.47
GMMREG	84.33 ± 19.15	48.19 ± 12.04	36.14 ± 9.00	42.94 ± 5.23	175.96 ± 38.87
MR-Net (Only L1)	**132.63 ± 30.49**	40.23 ± 15.55	**92.40 ± 19.86**	**70.12 ± 5.87**	85.81 ± 19.55
MR-Net (No PC feature)	128.33 ± 29.13	**38.88 ± 14.41**	89.44 ± 19.27	**70.05 ± 5.65**	**87.37 ± 19.16**
MR-Net (No 3D CNN)	80.70 ± 13.63	**38.99 ± 16.47**	41.71 ± 13.43	52.42 ± 17.37	56.40 ± 16.58
MR-Net	**131.69 ± 30.59**	**39.69 ± 12.71**	**92.00 ± 20.10**	**70.12 ± 4.33**	**88.36 ± 19.97**
MR-Net (automatic)	131.50 ± 30.81	**39.76 ± 12.81**	91.74 ± 20.28	**70.01 ± 4.54**	88.70 ± 20.15
GT Clinical Indices	132.24 ± 30.25	39.61 ± 11.92	92.63 ± 20.24	70.27 ± 3.88	87.78 ± 20.26

Table 3

Quantitative results for our MR-Net with incomplete data (with 2–5 slices and original input). The -2 slices and -4 slices denote the results with contours removing one/two pair of apical and basal slices.

Criterion	2 Slices	3 Slices	4 Slices	5 Slices	-2 Slices	-4 Slices	Original Input
CD (mm)	13.54 ± 14.65	7.94 ± 3.02	7.91 ± 3.36	6.97 ± 2.54	6.51 ± 1.98	9.97 ± 2.68	**5.22 ± 1.78**
EMD (mm)	12.17 ± 4.93	8.78 ± 2.65	8.74 ± 2.79	7.92 ± 2.31	7.74 ± 2.03	9.13 ± 2.40	**6.16 ± 1.75**
HD (mm)	11.83 ± 4.12	9.58 ± 2.83	9.34 ± 2.73	8.73 ± 2.40	9.00 ± 2.50	10.20 ± 2.89	**7.48 ± 1.96**
$\epsilon_{PC-to-PC}$ (mm)	5.46 ± 2.44	3.94 ± 1.11	3.88 ± 1.12	3.55 ± 0.96	3.46 ± 0.84	4.07 ± 1.36	**2.87 ± 0.73**

heart, and low signal-to-noise (SNR) ratio in parts of the myocardium, resulting in blurred boundaries for the left and right ventricles. Correspondingly, these errors are propagated to the manually or automatically extracted contours from such image volumes, which might cause missing contours at intermediate points across the heart. A 3D cardiac shape reconstruction framework robust to the presence of such irregularities, would be of significant clinical value as it would enable accurate quantification of cardiac functional indices, despite such artefacts. For that reason, the robustness of MR-Net to incomplete data in sparse 2D contours, used for 3D shape reconstruction, is also evaluated.

Incomplete samples are generated by retaining the basal and apical contours and randomly removing contours between. This process is used to generate four new samples with 2 to 5 slices each, for every sample in the original dataset. Additionally, to tackle the common issue encountered in routine CMR imaging, of missing apical/basal slices, two new samples with one/two pairs of base and apical slices missing are also generated. The resulting dataset, comprising 42,000 training samples, is used to re-train MR-Net and evaluate its robustness to incomplete data.

The quantitative and qualitative results in Table 3 and Fig. 4, respectively, indicate that MR-Net can generate accurate reconstructions of cardiac shape even in the presence of missing information (i.e. missing slices).

In the extreme scenario (reconstruction from 2 slices), only bottom and apical slices are given, our proposed MR-Net can still reconstruct high-quality meshes, although small misalignments exists between the reconstructed mesh and input contours. We observe that the reconstruction performance progressively improves with including more slices/contours, with a proportional decrease in the variance. When 5 slices are given for mesh reconstruction, the reconstruction performance is close to the results obtained using a complete stack of slices, across all metrics. Although the reconstruction accuracy of MR-Net for extreme scenarios is significantly lower than that of the original input (unmodified 2D contours), the values summarised in Table 3 indicate its performance is still comparable to/better than the baseline networks' performance on complete data (cf. Table 1).

To further evaluate the robustness of our approach, we employed the trained model to reconstruct meshes in the absence of apical and basal slices. As the apical and basal slices are essential to provide the network with contextual information regarding cardiac size, removing them significantly affects the quality of mesh reconstruction. Therefore, the results of removing apical and basal slices (-2 or -4 slices) are generally worse than removing the same number of slices between the apical and basal slices. However, our approach can still generate high-quality cardiac meshes with the basal/apical slices missing, as shown in Table 3 and Fig. 4.

The robustness of our approach to missing slices implies we can reconstruct high quality cardiac meshes using fewer annotated (manually/semi-automatically) slices and from sparse SAX cine-MR images. This provides avenues to reduce scan time in the future. Hence, the proposed approach could be of significant value in a clinical setting, especially for applications requiring real-time shape reconstruction (e.g. surgical navigation).

3.4. Shape reconstruction from autocontouring

To further validate the robustness and efficacy of MR-Net in realistic scenarios, in this section, we exemplify our method on shape reconstruction with contours extracted from automatic segmentation results instead of manual segmentations. Compared to manual segmentation, automatic segmentation results may contain several errors, posing a challenge for accurate 3D shape reconstruction. To be viable for a real clinical setting, however, a shape reconstruction method should be able to cope with such errors and facilitate accurate shape reconstruction from the original cardiac MR images.

For the samples in our testing dataset, the original MR images, the corresponding PCs of contours from manual segmentations and their target meshes are all available. Therefore, we use a deep learning-based cardiac segmentation method (Bai et al., 2018) to segment the original MR images, and then extract PCs of contours from the segmentation results. Finally, we apply our pre-trained MR-Net to reconstruct 3D cardiac meshes from them. As the target mesh for every MR image is available, we compare the predicted meshes with the former (shown in Table. 1 and Fig. 5).

As shown in Fig. 5, there are small differences between the input contours extracted from automatic segmentation results and manual segmentation results in terms of the number of contours, location and shape. However, even with those differences, our proposed MR-Net can still reconstruct accurate and high-quality meshes, achieving comparable performance to the reconstruction from manually segmented contours. This is further demonstrated by the results in Table. 1, where we see that mesh reconstruction accuracy using automatically segmented contours (MR-Net (au-
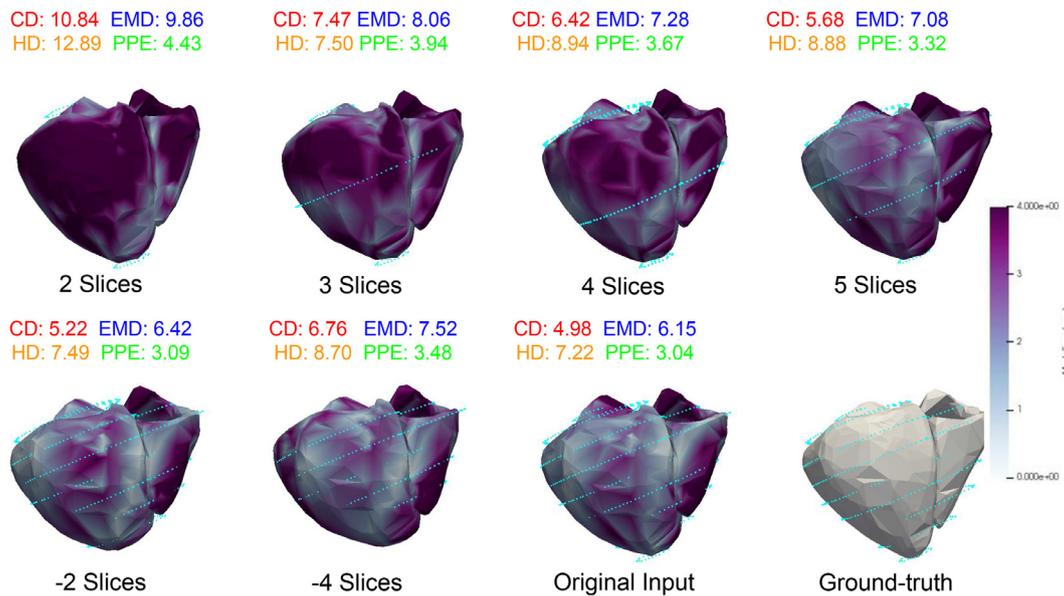
Fig. 4. 3D cardiac shape reconstruction with incomplete input. The CD, EMD, HD and PC-to-PC error (denoted by PPE in the figure) are shown on the left-top with red, blue, yellow and green numbers, respectively. The mesh colours indicate the PC-to-PC error from the predicted meshes to target meshes (colours corresponding to distance between 0.00 mm and 4.00 mm are shown in the colour bar).
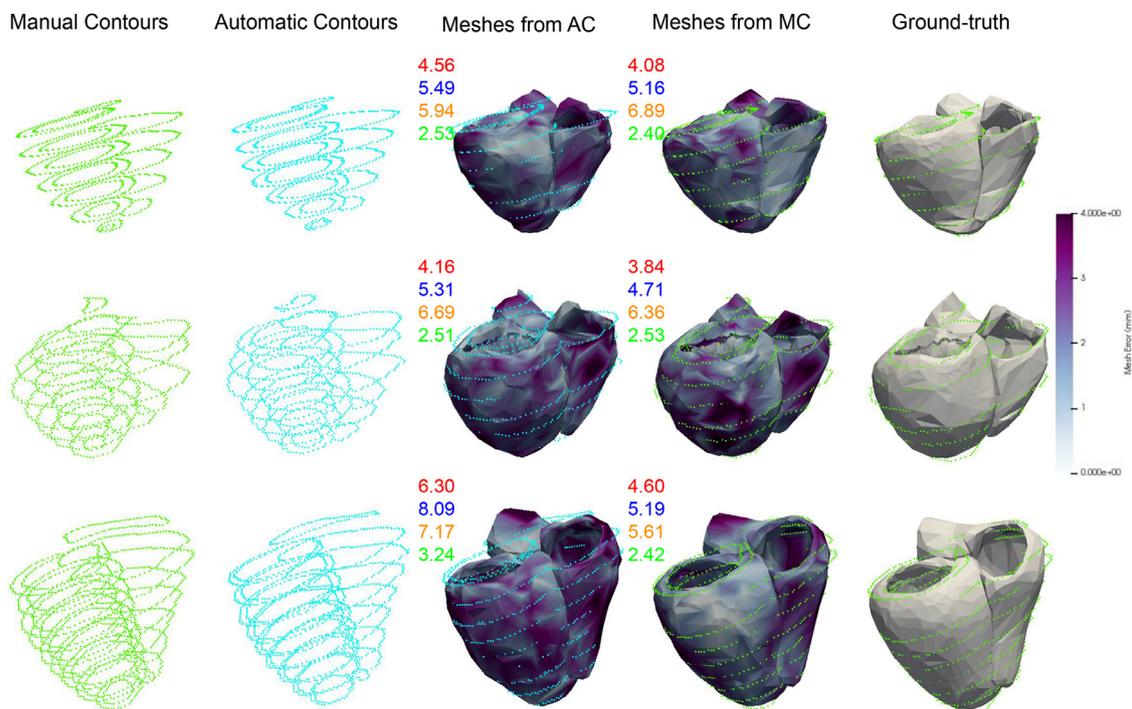
Fig. 5. Samples of 3D cardiac shape reconstruction using automatic and manual annotated contours. Each row is one sample. Columns from left to right are: Manual annotated contours (MC), automatic annotated contours (AC), reconstructed 3D meshes from both AC and MC, and ground-truth. The metrics (between predicted meshes and ground-truth) and predicted meshes are presented like in Fig. 4.

tomatic)) is a little worse than the results of mesh reconstruction from manual segmentation (MR-Net), but significantly better (achieving an average 3.5 mm PC-to-PC error about the ground-truth) than the other baseline networks investigated. During inference, MR-Net can reconstruct the shape of a sample is less than 0.1s on average, and $\sim$ 1s or less duration is required for the estimation of bi-ventricular contours using the deep learning-based segmentation method. Therefore, with the automatic segmentation method and MR-Net, we can reconstruct accurate, high-quality, 3D cardiac meshes from original cardiac MR images very

quickly ($\sim$ 1s), which is adequate for their use in real-time applications.

Compared with traditional 3D cardiac shape reconstruction approaches, MR-Net achieves a significant improvement in the inference time, without compromising the accuracy of the reconstructed 3D shapes. Additionally, as demonstrated, the proposed approach outperforms existing state-of-the-art deep learning approaches in terms of shape reconstruction accuracy. Assisted by deep learning-based segmentation methods, MR-Net can be further applied for the direct 3D shape reconstruction from original
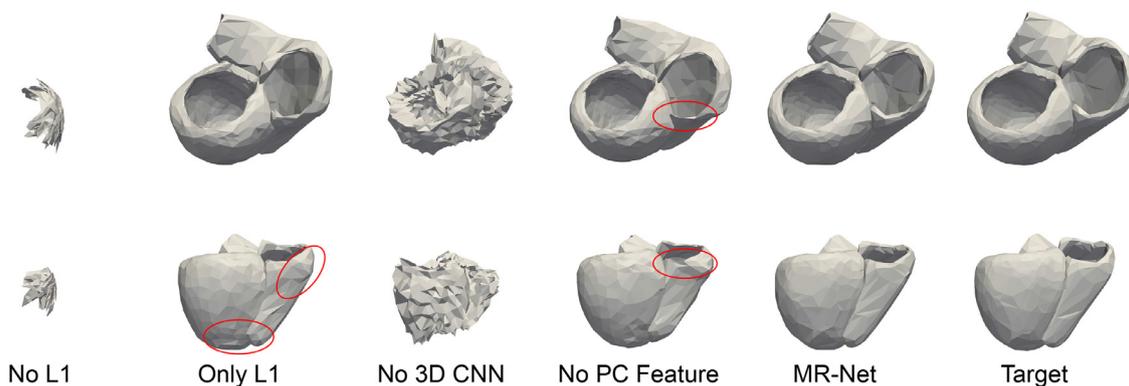
Fig. 6. The results predicted by different versions of MR-Net, where the first and second rows are the meshes from two different orientations.

MR/CT images. MR-Net can be applied to (1) guide other clinical image tasks in return (e.g. segmentation and registration) as it provides a continuous shape in 3D space, (2) in several real-time applications (e.g. surgical navigation), and (3) as an extension of clinical tools for visualising the 3D shape of anatomical structures. Although our proposed MR-Net can reconstruct highly similar meshes to the ground-truth, currently, the reconstruction accuracy is still constrained by the size of the 3D volume, which is the fundamental building block of PC-to-PC mapping. The reconstruction accuracy can be further improved with larger volume (e.g. $128 \times 128 \times 128$ or $256 \times 256 \times 256$ voxels) as the bridge for PC-to-PC mapping.

3.5. Ablation study

To analyse the contribution of different components in MR-Net, an ablation study is performed, as shown in Table 1, Table 2 and Fig. 6. MR-Net (No L1), MR-Net (Only L1), MR-Net (No PC feature), and MR-Net (No 3D CNN) refer to training MR-Net without the L1 loss, with just the L1 loss, without the PC feature extraction block, and without the 3D CNN feature extraction block, respectively. MR-Net achieves statistically significant improvements (evaluated using paired t-tests) to the aforementioned variations of MR-Net on all metrics ($p << 0.01$). Comparing the results between MR-Net (No L1) and MR-Net, we found that the L1 loss plays a key role in the network training, without which the network fails to reconstruct cardiac shapes. The other losses (except L1 loss) bring marginal improvements to the reconstruction accuracy, help better preserve fine structural details (viz. top and bottom of the right ventricle in Fig. 6) and facilitate generation of smoother meshes. Similarly, the lack of a PC feature extraction block weakens the reconstruction accuracy of MR-Net, while, lack of a 3D CNN feature extraction block significantly affects mesh reconstruction quality. Therefore, we can conclude that the L1 loss and 3D CNN feature extraction block are the key contributors to the reconstruction accuracy of MR-Net. The remaining components (other losses and the PC feature extraction block) help further refine mesh reconstruction accuracy.

4. Conclusion

A novel deep learning-based approach for 3D shape reconstruction from stacked 2D contours is proposed in this study. Our approach, MR-Net, can accurately reconstruct 3D shapes from sparse and incomplete 2D contour data, outperforming three state-of-the-art point cloud/mesh reconstruction networks. We further prove that our proposed approach can reconstruct accurate 3D cardiac meshes using contours generated by an automatic segmentation approach. This demonstrates that our model is robust to the segmentation errors induced by the latter. Using 2D automatic segmentation methods and our MR-Net, it is possible to reconstruct high-quality 3D cardiac meshes in real time. The versatile and robust nature of the proposed framework highlights its potential for application in several diagnostic and interventional settings. MR-Net is a supervised method, requiring ground-truth meshes during training. To alleviate the burden of curating high-quality ground truth meshes, which can be non-trivial in several applications, the problem of shape reconstruction from sparse contour/point cloud data can be tackled in an unsupervised manner. This could be achieved by approaching the problem in a manner similar to unsupervised deep learning-based image registration techniques, using the template mesh as the moving image and the point clouds of contours as the fixed image. This will be the subject of future work.

Declaration of Competing Interest

SEP provides consultancy to and is shareholder of Circle Cardiovascular Imaging, Inc., Calgary, Alberta, Canada.

CRediT authorship contribution statement

Xiang Chen: Conceptualization, Methodology, Software, Writing – original draft. **Nishant Ravikumar:** Supervision, Writing – review & editing. **Yan Xia:** Supervision, Visualization, Writing – review & editing. **Rahman Attar:** Conceptualization, Data curation. **Andres Diaz-Pinto:** Supervision, Writing – review & editing. **Stefan K Piechnik:** Writing – review & editing. **Stefan Neubauer:** Writing – review & editing. **Steffen E Petersen:** Writing – review & editing. **Alejandro F Frangi:** Supervision, Writing – review & editing.

References

Attar, R., Pereañez, M., Bowles, C., Piechnik, S.K., Neubauer, S., Petersen, S.E., Frangi, A.F., 2019. 3D Cardiac Shape Prediction with Deep Neural Networks: Simultaneous Use of Images and Patient Metadata. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 586–594.

Bai, W., Sinclair, M., Tarroni, G., Oktay, O., Rajchl, M., Vaillant, G., Lee, A.M., Aung, N., Lukaschuk, E., Sanghvi, M.M., et al., 2018. Automated cardiovascular magnetic resonance image analysis with fully convolutional networks. J. Cardiovasc. Magn. Reson. 20 (1), 65.

Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P., 2017. Geometric deep learning: going beyond euclidean data. IEEE Signal Process Mag 34 (4), 18–42.

Defferrard, M., Bresson, X., Vandergheynst, P., 2016. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In: Advances in Neural Information Processing Systems, pp. 3844–3852.

Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., Yuan, J., 2019. 3D hand shape and pose estimation from a single RGB image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 10833–10842.

Hashimoto, T., Saito, M., 2019. Normal estimation for accurate 3D mesh reconstruction with point cloud model incorporating spatial structure. In: CVPR Workshops, pp. 54–63.

Jian, B., Vemuri, B.C., 2011. Robust point set registration using gaussian mixture models. IEEE Trans Pattern Anal Mach Intell 33 (8), 1633–1645.

Jiang, H., Cai, J., Zheng, J., 2019. Skeleton-aware 3D human shape reconstruction from point clouds. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5431–5441.

Kipf, T.N., Welling, M., 2017. Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations.

Kolotouros, N., Pavlakos, G., Daniilidis, K., 2019. Convolutional mesh regression for single-image human shape reconstruction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4501–4510.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.

Lehmann, H., Kneser, R., Neizel, M., Peters, J., Ecabert, O., Kühl, H., Kelm, M., Weese, J., 2009. Integrating viability information into a cardiac model for interventional guidance. In: International Conference on Functional Imaging and Modeling of the Heart. Springer, pp. 312–320.

Li, R., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A., 2019. PU-GAN: a point cloud upsampling adversarial network. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 7203–7212.

Lim, C.W., Su, Y., Yeo, S.Y., Ng, G.M., Nguyen, V.T., Zhong, L., San Tan, R., Poh, K.K., Chai, P., 2014. Automatic 4D reconstruction of patient-specific cardiac mesh with 1-to-1 vertex correspondence from segmented contours lines. PLoS ONE 9 (4), e93747.

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J., 2015. SMPL: a skinned multi-person linear model. ACM Trans. Graph. 34 (6), 1–16.

Lorensen, W.E., Cline, H.E., 1987. Marching cubes: a high resolution 3D surface construction algorithm. ACM Siggraph Comput. Graph. 21 (4), 163–169.

Medrano-Gracia, P., Cowan, B.R., Bluemke, D.A., Finn, J.P., Lima, J.A.C., Suinesiaputra, A., Young, A.A., 2013. Large scale left ventricular shape atlas using automated model fitting to contours. In: International Conference on Functional Imaging and Modeling of the Heart. Springer, pp. 433–441.

Myronenko, A., Song, X., 2010. Point set registration: coherent point drift. IEEE Trans. Pattern Anal. Mach. Intell. 32 (12), 2262–2275.

Pan, J., Han, X., Chen, W., Tang, J., Jia, K., 2019. Deep mesh reconstruction from single RGB images via topology modification networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 9964–9973.

Petersen, S.E., Aung, N., Sanghvi, M.M., Zemrak, F., Fung, K., Paiva, J.M., Francis, J.M., Khanji, M.Y., Lukaschuk, E., Lee, A.M., et al., 2017. Reference ranges for cardiac structure and function using cardiovascular magnetic resonance (CMR) in caucasians from the UK biobank population cohort. J. Cardiovasc. Magn. Reson. 19 (1), 18.

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017. PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 652–660.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, pp. 5099–5108.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations.

Suinesiaputra, A., Ablin, P., Alba, X., Alessandrini, M., Allen, J., Bai, W., Cimen, S., Claes, P., Cowan, B.R., D'hooge, J., et al., 2017. Statistical shape modeling of the left ventricle: myocardial infarct classification challenge. IEEE J. Biomed. Health Inform. 22 (2), 503–515.

de Vila, M.H., Attar, R., Pereañez, M., Frangi, A.F., 2018. MULTI-X, a state-of-the-art cloud-based ecosystem for biomedical research. In: IEEE International Conference on Bioinformatics and Biomedicine. IEEE, pp. 1726–1733.

Villard, B., Grau, V., Zacur, E., 2018. Surface mesh reconstruction from cardiac MRI contours. J. Imaging 4 (1), 16.

Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.-G., 2018. Pixel2mesh: Generating 3D Mesh Models from Single RGB Images. In: Proceedings of the European Conference on Computer Vision, pp. 52–67.

Wang, Z.-Y., Zhou, X.-Y., Li, P., Theodoreli-Riga, C., Yang, G.-Z., 2020. Instantiation-Net: 3D mesh reconstruction from single 2D image for right ventricle. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 680–691.

Wen, C., Zhang, Y., Li, Z., Fu, Y., 2019. Pixel2mesh++: multi-view 3D mesh generation via deformation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1042–1051.

Xu, H., Zacur, E., Schneider, J.E., Grau, V., 2019. Ventricle surface reconstruction from cardiac MR slices using deep learning. In: International Conference on Functional Imaging and Modeling of the Heart. Springer, pp. 342–351.

Yu, L., Li, X., Fu, C.-W., Cohen-Or, D., Heng, P.-A., 2018. PU-Net: point cloud upsampling network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2790–2799.

Zhou, X.-Y., Wang, Z.-Y., Li, P., Zheng, J.-Q., Yang, G.-Z., 2019. One-stage shape instantiation from a single 2D image to 3D point cloud. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 30–38.

Zou, M., Holloway, M., Carr, N., Ju, T., 2015. Topology-constrained surface reconstruction from cross-sections. ACM Trans. Graph. 34 (4), 1–10.