Neural Deformable Models for 3D Bi-Ventricular Heart Shape Reconstruction and Modeling from 2D Sparse Cardiac Magnetic Resonance Imaging

Meng Ye[1], Dong Yang[2], Mikael Kanski[3], Leon Axel[3], Dimitris Metaxas[1]
[1]Rutgers University, [2]NVIDIA, [3]New York University School of Medicine
{my389, dnm}@cs.rutgers.edu

Abstract

We propose a novel neural deformable model (NDM) targeting at the reconstruction and modeling of 3D bi-ventricular shape of the heart from 2D sparse cardiac magnetic resonance (CMR) imaging data. We model the bi-ventricular shape using blended deformable superquadrics, which are parameterized by a set of geometric parameter functions and are capable of deforming globally and locally. While global geometric parameter functions and deformations capture gross shape features from visual data, local deformations, parameterized as neural diffeomorphic point flows, can be learned to recover the detailed heart shape. Different from iterative optimization methods used in conventional deformable model formulations, NDMs can be trained to learn such geometric parameter functions, global and local deformations from a shape distribution manifold. Our NDM can learn to densify a sparse cardiac point cloud with arbitrary scales and generate high-quality triangular meshes automatically. It also enables the implicit learning of dense correspondences among different heart shape instances for accurate cardiac shape registration. Furthermore, the parameters of NDM are intuitive, and can be used by a physician without sophisticated post-processing. Experimental results on a large CMR dataset demonstrate the improved performance of NDM over conventional methods.

1. Introduction

Cardiac magnetic resonance imaging (CMR) is the gold standard for non-invasive evaluation of global cardiac function, i.e., blood pumping of the left ventricle (LV). Due to the relatively slow imaging speed of MR, current clinical CMR scan protocols are 2D-based. To recover the 3D geometry of the LV and right ventricle (RV), we usually scan a stack of short axis (SAX) and several long axis (LAX) images. As shown in Fig. 1 (a-f), SAX images cover the range from the base to the apex of the heart and LAX images in-

clude 2-, 3-, 4-chamber views. Although in-plane resolution of the image is high enough to capture 2D cardiac anatomical details, the through-plane resolution is much lower, in order to reduce imaging time. Therefore, the SAX and LAX images together can only produce a sparse 3D point cloud of the heart, as shown in Fig. 1 (g). The dense and accurate 3D geometry reconstruction, as shown in Fig. 1 (h), is thus needed for not only the downstream image analysis tasks, such as the estimation of LV mass and volume[40], and 3D cardiac wall strain fields [26, 17, 19], but also other clinical applications, such as image-guided interventions [32] and biomechanics finite element-based simulations [39, 37, 16].

3D cardiac geometry reconstruction has a long history [24, 26, 40, 23, 38, 8, 31, 20]. Most of the previous methods reconstruct the 3D shape from a sparse point cloud generated from segmentation results on CMR images. Those segmentations delineate the myocardium wall, blood pool of the LV and RV, as shown by the green points in Fig. 1. As one of the conventional methods, template mesh adaptation first constructs a template mesh that describes the mean shape of the target and then registers this template with the sparse point cloud [38]. The resulting meshes are usually not accurate for data that have large shape variation from the mean shape, which leads to a generalization problem of this kind of approach.

We are inspired by the success of conventional deformable models (CDMs) for their efficiency in shape reconstruction and ability to provide intuitive and explainable shape parameters [26, 28, 23]. CDMs can efficiently describe the target shape using a set of global and local deformation parameters. However, to fit a shape primitive to a target shape, conventional methods use iterative optimization and fit the primitive to the given data. We propose a neural deformable model (NDM), which is composed of global deformations using parameter functions and local deformations using neural diffeomorphic flows. It can be trained to learn the global and local deformation parameters, conditioning on the sparse given data, from a shape distribution manifold. As shown in Fig. 1 (h), NDM can accurately reconstruct the bi-ventricle shape from the
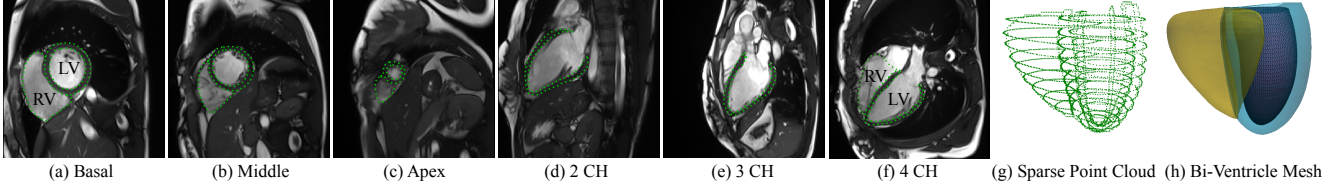
Figure 1. Cardiac MR standard scan views and bi-ventricular shape reconstruction. (a-c) Short axis (SAX) views at the basal/middle/apex region of the left ventricle (LV). (d-f) Long axis (LAX) 2/3/4 chamber (CH) views. Green points show the endo-/epi-cardial border of LV and the border of right ventricle (RV). (g) Sparse cardiac point cloud with 10 SAX and 3 LAX slices' data. (h) Bi-ventricular mesh reconstructed from (g) by our neural deformable model, in which the LV endo-cardial, epi-cardial and RV surfaces are shown in red, blue and yellow, respectively.

sparse point cloud, as shown in Fig. 1 (g). NDM also enables implicit learning of dense correspondences among shape instances, which facilitates accurate shape registration. Lastly, NDM estimates shape parameters of the heart directly, which can be used by clinicians without sophisticated post-processing.

Our key contributions in this work can be summarized as follows: (1) We formulate a new shape modeling framework which can capture global shape information and recover complex shape details accurately; (2) We design a novel neural deformable model and propose an efficient coarse-to-fine learning paradigm to learn the global and local deformation parameters from a shape distribution manifold; (3) We show how such a shape modeling framework can be trained to perform clinically meaningful 3D heart shape reconstruction, registration and interpretation. Furthermore, our method has the potentials to inspire shape reconstruction and registration tasks in other domains.

2. Related Work

2.1. 3D Heart Ventricle Reconstruction

Most existing 3D heart ventricle shape reconstruction methods are based on template mesh registration. They aim to register a template mesh which has a dense set of vertices to a sparse input point cloud. Early works [24, 26, 40, 23, 38] use finite element shape models to take advantage of basis function interpolation and extrapolation and perform a least square fitting procedure to fit the initial mesh to the sparse visual data. The fitting procedure is a time-consuming iterative process. A recent learning-based template mesh registration work [8] can learn to reconstruct bi-ventricular shape from ground truth shape data, enabling the usage of not only observed sparse visual data but also a shape distribution manifold. However, the low resolution of its 3D binary volume, used as the bridge between the input and template for shape feature transferring, decreases the 3D shape reconstruction accuracy. The main drawback of the template mesh registration method is the lack of generalization, when the shape variation between the template

and the target is too large to accurately register them. Our method decomposes the geometry correspondence between the shape primitive and the target into a set of global and local deformation parameters, which are learned in a coarse-to-fine fashion, resulting in accurate shape reconstructions even in the presence of large variations between the shape primitive and the target.

2.2. Deformable Models

Deformable models [21, 34, 36, 35, 10] are in the form of 2D curves, 3D surfaces or volumes whose points are parameterized in the material coordinate domain. Such models, also called deformable primitives, are deformed under the effect of external and internal forces. External forces are computed from the visual data, to drive the model to fit the target data. Internal forces are defined within the model itself, to preserve its smoothness during deformation. In [27, 26], constant geometric parameters of a primitive are extended to parameter functions to capture local shape variations. In our work, due to the aforementioned advantage, we use this parameter function-based primitive representation and combine it with a neural diffeomorphic flow [25, 15, 33] for local deformations, to further improve shape reconstruction accuracy.

2.3. Diffeomorphic Flow

Diffeomorphic flow is a spatial transformation that is smooth and whose inverse is smooth. In computational anatomy modeling of the brain, large deformation diffeomorphic metric mapping (LDDMM) has been used to compute a time-dependent velocity vector field based on an ordinary differential equation (ODE) [3]. To overcome the computation complexity of LDDMM, a stationary velocity field (SVF) is introduced to parameterize diffeomorphisms [1]. Recent work on neural ordinary differential equations (NODE) [7, 6] enables the solution of neural diffeomorphic flow [25, 15, 33]. We propose to integrate parameter function-based primitive and neural diffeomorphic flow parameterized local deformation as a novel neural deformable model, which enables the efficient coarse-to-fine
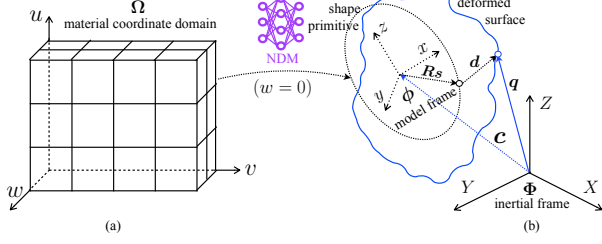
2

Figure 2. Geometry representation of a deformable surface model. (a) Material coordinate domain $\mathbf{\Omega}$. (b) The shape primitive and deformed surface mapped from the material coordinate domain (illustrated with $w = 0$) by a neural deformable model (NDM).
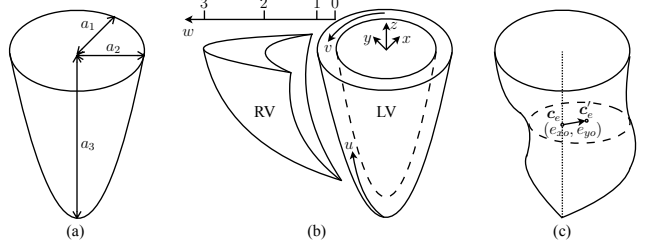

Figure 3. (a) A deformable shape primitive with aspect ratios a_1, a_2, a_3. (b) Blended deformable shape primitives for the modeling of bi-ventricular geometry. (c) Axis offset deformations e_{xo}, e_{yo} that translate the center of the ellipse from $\boldsymbol{c}_e$ to $\boldsymbol{c}'_e$.

learning of its geometric and deformation parameters from a shape distribution manifold.

3. Method

In this section, we introduce a neural deformable model (NDM) for bi-ventricle shape reconstruction, mesh generation and shape registration. We first provide the formulation of blended deformable superquadrics and then introduce a learning algorithm that can be applied to the NDM.

3.1. General Geometry and Parameter Functions-based Deformable Models

As shown in Fig. 2, we formulate a 3D object surface model in space using material coordinates $\boldsymbol{m} = (u, v, w)$, which are defined on a domain $\mathbf{\Omega}$. Given an inertial frame of reference $\mathbf{\Phi}$ in 3D, the positions of points $\boldsymbol{q}$ on the surface model relative to $\mathbf{\Phi}$ are represented by the following vector-valued function of $\boldsymbol{m}$:

$$\boldsymbol{q}(\boldsymbol{m}) = (x(\boldsymbol{m}), y(\boldsymbol{m}), z(\boldsymbol{m}))^\top , \quad (1)$$

where $\top$ is the transpose operator. To model the object pose, we introduce a model-centered reference frame ϕ and represent the surface points positions as

$$\boldsymbol{q} = \boldsymbol{c} + \boldsymbol{R}\boldsymbol{s} + \boldsymbol{d}, \quad (2)$$

where $\boldsymbol{c}$ is the origin of model frame ϕ, $\boldsymbol{R}$ is the rotation matrix[1] that describes the orientation of ϕ, $\boldsymbol{s}$ is a shape primitive defined in the model frame ϕ, and $\boldsymbol{d}$ is a local deformation function.

A shape primitive $\boldsymbol{s}$ could be a generalized cylinder [14], geon [4], hyperquadric [18], or superquadric [35]. To model the bi-ventricular shape, we pick a specific kind of superquadric, an ellipsoid, as our primitive:

$$\boldsymbol{e}_e = a_0 \begin{pmatrix} a_1 \cos u \cos v \\ a_2 \cos u \sin v \\ a_3 \sin u \end{pmatrix}, \quad (3)$$

---

[1]We use a quaternion with unit magnitude to represent it.

where $-\pi/2 \leq u \leq \alpha_w$, $-\pi \leq v < \pi$, $a_0 > 0$, $0 \leq a_1, a_2, a_3 < 1$. Specifically, a_0 is a scale parameter, and a_1, a_2 and a_3 are the aspect ratio parameters along the x-, y- and z-axes, respectively. We show an example ellipsoid in Fig. 3 (a). To create a deformable shape primitive with more intuitive deformation degrees of freedom, we replace the constant parameters in a superquadric ellipsoid with *parameter functions* [27] of u, w:

$$\boldsymbol{e} = \boldsymbol{e}(\boldsymbol{m}; a_0(w), a_1(u, w), a_2(u, w), a_3(u, w))$$
$$= a_0(w) \begin{pmatrix} a_1(u, w) \cos u \cos v \\ a_2(u, w) \cos u \sin v \\ a_3(u, w) \sin u \end{pmatrix} . \quad (4)$$

3.2. Shape Blending for Bi-Ventricular Geometry

As shown in Fig. 3 (b), the LV endo- and epi-cardial surfaces $\boldsymbol{s}(u, v, w = 0, 1)$ could be modeled by a two-layer deformable shape, defined in Eq. (4), with $w = 0, 1$. Considering the significant shape difference of RV from LV, we use *a blended shape* to model the RV. A single blended shape is the combination of two component shape primitive parts [10]. We cut out portions of component primitives and join the remaining portions together:

$$\boldsymbol{s}(u, v, 2) = \begin{cases} \boldsymbol{e}((u, v, 2); ., ., a_2(u, 2), .) & if \ 0 \leq v < \pi \\ \boldsymbol{e}((u, -v, 3); ., ., a_2(u, 3), .) & if \ -\pi \leq v < 0 \end{cases},$$
$$(5)$$

where $\boldsymbol{e}(u, v, 2)$ and $\boldsymbol{e}(u, v, 3)$ only differ by $a_2(u)$. In this way, we model the bi-ventricular geometry as blended surfaces $\boldsymbol{s}(u, v, w)$ with $w = 0, 1, 2$, $\alpha_0 = \alpha_1 = \pi/6$, $\alpha_2 = \pi$.

Although (geometry) parameter functions greatly increase the capturing capacity of shape features beyond constant parameter-based superquadrics, we can further apply global deformations to the underlying shape primitive $\boldsymbol{e}$ with continuous deformation parameter functions. Here, we apply axis offset deformations $\mathbf{T}_o$ to $\boldsymbol{e}$:

$$\boldsymbol{s} = \mathbf{T}_o(\boldsymbol{e}; e_{xo}(u), e_{yo}(u)) = \begin{pmatrix} e_x + e_{xo}(u, w) \\ e_y + e_{yo}(u, w) \\ e_z \end{pmatrix}, \quad (6)$$

3

where $e_{xo}(u, w)$, $e_{yo}(u, w)$ are axis-offset parameter functions along the x- and y- axes, respectively. We show an example of $\mathbf{T}_o$ deformed ellipsoid in Fig. 3 (c). It is obvious that $\mathbf{T}_o$ describes the bending deformation along the long axis of the heart ventricle. Note that, although the parameter functions are defined with local u, they are of "global" deformation effect on each u. Therefore, we call them global parameters in our work.

3.3. Diffeomorphic Point Flow for Local Deformations

In Eq. (2), we use local deformation $\boldsymbol{d}$ to describe details of complex shapes. In our NDM, local deformation is modeled as diffeomorphic point flows. Let $\mathcal{D}(\boldsymbol{q}, t) : \mathcal{V} \subset \mathbb{R}^{N \times 3} \times [0, 1] \mapsto \mathcal{V} \subset \mathbb{R}^{N \times 3}$ be a time $t \in [0, 1]$ parameterized spatial mapping such that, for N points $\boldsymbol{q} = \boldsymbol{q}^{(0)} = \mathcal{D}(\boldsymbol{q}, 0)$, $\boldsymbol{q}' = \boldsymbol{q}^{(1)} = \mathcal{D}(\boldsymbol{q}, 1)$ are the final transformed points, and $\boldsymbol{d} = \mathcal{D}(\boldsymbol{q}, 1) - \mathcal{D}(\boldsymbol{q}, 0)$. A diffeomorphic mapping is achieved as the trajectory and integration of a smooth *velocity field* $\boldsymbol{v} : \mathcal{V} \subset \mathbb{R}^{N \times 3} \times [0, 1] \mapsto \mathcal{V} \subset \mathbb{R}^{N \times 3}$, which is governed by an ODE, also known as the flow equation [12], as follows:

$$\frac{\partial \mathcal{D}(\boldsymbol{q}, t)}{\partial t} = \boldsymbol{v}(\mathcal{D}(\boldsymbol{q}, t), t) \quad s.t. \quad \mathcal{D}(\boldsymbol{q}, 0) = \boldsymbol{q}. \quad (7)$$

The initial value problem (IVP) in Eq. (7) could be solved with a neural ODE solver [7] with the dynamic function the velocity field. According to the *Cauchy-Lipschitz* theorem [5], if the velocity field is Lipschitz continuous, the resulting transform $\mathcal{D}$ is a *bi-Lipschitz map*, which is also a diffeomorphism in essence [11].

In this work, we consider the pose ($\boldsymbol{c}$ and $\boldsymbol{R}$) and shape parameter functions in the primitive $\boldsymbol{s}$ as generalized deformations. Therefore, our neural deformable model consists of the following deformation parameter vector:

$$\boldsymbol{q}_N = (\boldsymbol{q}_g^\top, \boldsymbol{q}_d^\top)^\top, \quad (8)$$

where $\boldsymbol{q}_g = (\boldsymbol{c}^\top, \boldsymbol{R}^\top, \boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3, \boldsymbol{e}_{xo}, \boldsymbol{e}_{yo})^\top$ is the global deformation parameter vector and $\boldsymbol{q}_d = \boldsymbol{d}$ is the local deformation parameter vector.

3.4. Architecture of NDM

We learn $\boldsymbol{q}_N$ for each shape instance in a coarse-to-fine fashion: we first learn $\boldsymbol{q}_g$ and then we learn $\boldsymbol{q}_d$. As shown in Fig. 4, our NDM has three branches that predict each $\boldsymbol{q}_N$ of the LV endo-, epi-cardial surfaces and the RV surface, respectively. We use a shared point transformer (PT) encoder and three point transformer decoders to learn shape embeddings from a given sparse point cloud. The PT architecture is the same as in [41] for semantic segmentation, except that we replace its last multi-layer perceptron (MLP) layer with a global average pooling (GAP) layer to get global shape

embedding $\boldsymbol{z}$. With $\boldsymbol{z}$, we first use MLPs to predict the global deformation parameter vector $\boldsymbol{q}_g$. The shape primitive $\boldsymbol{s}$ is then deformed by $\boldsymbol{q}_g$:

$$\boldsymbol{s}_g = \boldsymbol{s} \circ \boldsymbol{q}_g, \quad (9)$$

where $\circ$ describes global deformations, and $\boldsymbol{s}_g$ is the resulting globally deformed shape primitive, which captures the coarse target shape features, as shown in Fig. 4. As we will illustrate next, during training of NDM, we exploit a marginal space learning method [42] to enforce the global deformations to account for as much of the target shape as possible. Then we use a *conditional* diffeomorphic point flow generation block $\mathcal{D}$ to learn local deformations:

$$\frac{\partial \mathcal{D}(\boldsymbol{q}; \boldsymbol{z}, t)}{\partial t} = \boldsymbol{v}(\mathcal{D}(\boldsymbol{q}, t); \boldsymbol{z}, t) \quad s.t. \quad \mathcal{D}(\boldsymbol{q}; \boldsymbol{z}, 0) = \boldsymbol{s}_g. \quad (10)$$

We call $\mathcal{D}$ the neural ordinary equation (NODE) block. We use the same NODE block as in [15], except that we exclude its instance normalization layer. By solving Eq. (10), we get the final reconstructed shape $\boldsymbol{Q}' = \mathcal{D}(\boldsymbol{q}; \boldsymbol{z}, 1)$. From Fig. 4, the local deformations can further refine target shape details. As such, our coarse-to-fine architecture of NDM results in accurate target shape reconstructions.

3.5. Triangular Mesh Generation

Both global and local deformations in $\boldsymbol{q}_N$ are smooth with inverse smooth. Therefore, $\boldsymbol{q}_N$ is a diffeomorphic mapping and the final shape reconstruction result $\boldsymbol{Q}'$ preserves the topology of shape primitive $\boldsymbol{s}$. Since $\boldsymbol{s}$ is an ellipsoid, we can get the corresponding mesh by connecting any three nearest-neighboring surface vertices. We then take the edges of this ellipsoid mesh as those of our target mesh. In this way, we can automatically reconstruct a triangular mesh $\mathcal{M}$ from $\boldsymbol{Q}'$.

3.6. Shape Registration

NDM implicitly learns dense correspondences between different shape instances, which can be used to register shapes. As shown in Fig. 5, we first use NDM to fit the primitive $\boldsymbol{s}$ to the targets $\mathcal{M}_1$ and $\mathcal{M}_2$. The dense correspondences learned by NDM will map a point q of the primitive to q_1 and q_2 for $\mathcal{M}_1$ and $\mathcal{M}_2$, respectively. Obviously, q_1 may not coincide with the point p_1 of $\mathcal{M}_1$. For each point p_1 of $\mathcal{M}_1$, we first look up the nearest neighbouring point q_1 reconstructed by NDM, then we map p_1 to q_2 via the correspondence $p_1 \rightarrow q_1 \rightarrow q \rightarrow q_2$. In this way, we can register $\mathcal{M}_1$ to $\mathcal{M}_2$.

3.7. Model Training

We use the Chamfer distance (CD) loss [13] to encourage the geometric similarity between vertices $\boldsymbol{Q}'$ of the reconstructed mesh and the ground truth point cloud $\mathcal{Q}$:

$$\mathcal{L}_{geo} = \mathcal{L}_{CD}(\boldsymbol{Q}', \mathcal{Q}). \quad (11)$$
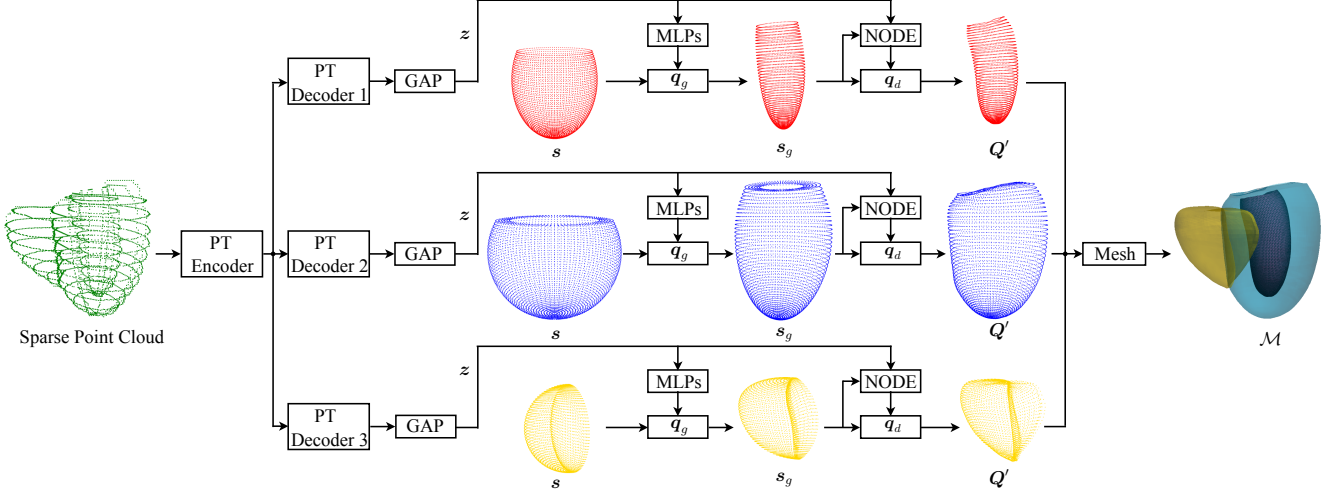
Figure 4. Neural deformable models (NDMs) for bi-ventricular cardiac point cloud upsampling and automated triangular mesh generation. Each deformation parameter vector $\boldsymbol{q}_N = (\boldsymbol{q}_g^\top, \boldsymbol{q}_d^\top)^\top$ of a NDM is learned in a coarse-to-fine fashion, from the learning of global deformation ($\boldsymbol{q}_g$) to local deformation ($\boldsymbol{q}_d$). $\boldsymbol{q}_g$ captures gross target shape features, leaving only small shape details to be refined by $\boldsymbol{q}_d$.
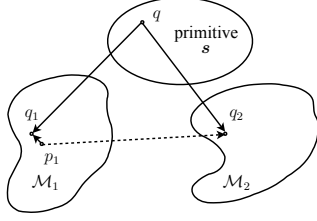

Figure 5. Shape registration via the implicitly learned dense correspondence by NDM.

We also include two regularization losses. The first one is to regularize the amount of local deformations:

$$\mathcal{L}_d = \|\boldsymbol{q}_d\|_2^2, \tag{12}$$

where $\boldsymbol{q}_d = \boldsymbol{Q}' - \boldsymbol{s}_g$ is the local deformation field. The second one is to regularize the smoothness of the local deformation field:

$$\mathcal{L}_s = \|\bigtriangledown \boldsymbol{q}_d\|_2^2, \tag{13}$$

where $\bigtriangledown \boldsymbol{q}_d$ is the gradients of $\boldsymbol{q}_d$ over the material coordinates u and v. In summary, the total loss function is the weighted sum of the geometric similarity loss and the local deformation regularization terms:

$$\mathcal{L} = \mathcal{L}_{geo} + \lambda_d \mathcal{L}_d + \lambda_s \mathcal{L}_s, \tag{14}$$

where λ_d and λ_s are the weighting hyper-parameters.

The learning of $\boldsymbol{q}_N$ is to search the optimum deformation parameters. The search space composed of $\boldsymbol{q}_N$ is so large that end-to-end training by optimizing Eq. (14) leads to slow convergence. We adopt the marginal space learning (MSL) method [42, 30] to train NDM. More specifically, we decompose the training process into a chain of sub-processes, in which we gradually add one component of the deformation parameter vector $\boldsymbol{q}_N$ into NDM at a time. In this way, each training sub-process only focuses on one individual deformation component, to speed up the convergence.

4. Experiments

4.1. Dataset and Pre-Processing

We use a large public 3D CMR dataset [2] of $1,331$ normal subjects to evaluate our method. Each subject contains the end-diastole (ED) and end-systole (ES) phases. Both the low resolution (LR) and high resolution (HR) segmentation masks of each subject are provided in this dataset. We took the HR data, which was scanned by 3D high spatial resolution of $1.25 \times 1.25 \times 2~mm^3$ CMR protocols [9] as our experimental dataset. We extracted vertices from the bi-ventricular mesh reconstructed from the segmentation mask volume to form a dense ground truth point cloud. Next, we generated a sparse point cloud as the input by mimicking standard clinical 2D CMR scanning. We first determined the LAX and SAX view planes. Then we sliced the segmentation mask volume at these planes and extracted points along the segmentation contours. We downsampled these points using farthest point sampling (FPS) [29] to a fixed number. We sampled 3 LAX and 10 SAX planes and we set the input point number as $5,600$. We randomly split the dataset into 900, 200 and 231 subjects as the training, validation and test sets, respectively. For the pre-processing, we centered each input point cloud at $(0, 0, 0)$ by subtracting the center coordinates and linearly normalized x, y, z

coordinates to $[-0.85, 0.85]$. We also rotated each data set to align the center line of LV and RV with the y-axis.

4.2. Evaluation Metrics

For quantitative evaluation of the geometric similarity, we followed [8, 22] to use Chamfer distance (CD), earth mover's distance (EMD) and point-to-surface distance (P2F) to measure the geometric similarity between the densified point cloud and the ground truth. For the evaluation of generated mesh quality, we followed [33] to compute the normal consistency (NC), easy non-manifold face (ENF) ratio (with $\delta = 0$), and self-intersection (SI) ratio. We also used CD, EMD, P2F to measure the shape registration accuracy. The lower the evaluation metric, the better the performance, except for NC, which is the opposite. Note that we do all the quantitative evaluation in the original data space, not in the normalized data space.

4.3. Baseline Methods

We compared our method with a state-of-the-art (SOTA) 3D bi-ventricular geometry reconstruction method MR-Net [8], and a SOTA 3D manifold mesh generation method NMF [15]. For MR-Net, we first randomly sampled a data set from the training dataset; then we used our method to fit the primitive to the ground truth shape; the reconstructed shape was used as the template mesh. For NMF, the template spheres were replaced by the same initial shape primitives as ours. We used their publicly available codes and followed the same setting of the original work to train their networks with our training dataset. Note that we computed the geometric similarity losses of LV-endo, LV-epi and RV surfaces separately for the training of our method, MR-Net and NMF. We also trained our model in an unsupervised fashion (Ours-un), with the input sparse point cloud as the ground truth, which is similar to conventional iterative optimization-based deformable models. We compared our accuracy of shape registration based on the learned dense correspondence with MR-Net, NMF and Ours-un.

4.4. Implementation Details

We implemented NDM with PyTorch. And we set the hyper-parameters $\lambda_d = 0.1$, $\lambda_s = 0.05$. Adam optimizer is used with a learning rate of $5e^{-4}$ to train our model with a batch size of 2. During training, we sampled $5,000$, $5,500$ and $5,000$ points from the reconstructed LV-endo, LV-epi and RV meshes, respectively. All models were trained on an NVIDIA Quadro RTX 8000 GPU. For evaluation, we uniformly sampled the same number of $3,000$ points, on the predicted mesh as the ground truth for the calculation of CD, EMD and P2F for the LV-endo, LV-epi and RV surfaces, respectively.

Phase	Method	CD ↓	EMD ↓	P2F ↓	NC ↑	ENF ↓	SI ↓
	MR-Net	3.09	6.05	1.47	0.762	1.48	0.035
	NMF	6.31	6.50	4.40	0.759	1.46	0.019
ED	Ours-un	5.16	6.20	3.11	0.751	**1.45**	0.002
	Ours	**2.73**	**5.80**	**1.17**	**0.765**	1.46	**0**
	MR-Net	2.37	4.50	1.23	0.750	1.47	0.068
	NMF	3.74	4.74	2.46	0.752	1.46	0.027
ES	Ours-un	2.90	4.48	1.63	0.747	**1.45**	0.002
	Ours	**1.91**	**4.07**	**0.873**	**0.758**	1.46	**0**

Table 1. Mean Chamfer distance (CD) (mm), earth mover's distance (EMD) (mm) and point-to-surface distance (P2F) (mm) for geometric similarity evaluation. Mean normal consistency (NC), easy non-manifold face (ENF) ratio and self-intersection (SI) ratio for cardiac mesh quality evaluation.

Method	CD (mm) ↓	EMD (mm) ↓	P2F (mm) ↓
MR-Net	2.44	4.74	1.33
NMF	3.76	4.92	2.45
Ours-un	3.03	4.41	1.91
Ours	**1.94**	**3.95**	**1.00**

Table 2. Mean Chamfer distance (CD), earth mover's distance (EMD) and point-to-surface distance (P2F) for shape registration.

4.5. Results

4.5.1 Shape Reconstruction Performance

In Table 1, we show the mean quantitative results of bi-ventricle shape reconstruction for ED and ES phases. More detailed quantitative results of LV-endo, LV-epi and RV surface reconstruction are presented in Supplementary Material. We also show an example in Fig. 6. Our method significantly outperforms baseline methods for both geometric similarity and mesh quality aspects. MR-Net cannot handle large shape variations between the template mesh and the target, resulting in significant shape artefacts in the reconstructions. Such artefacts could produce self-intersected local surfaces, as demonstrated by the SI value. NMF utilizes multiple deformation blocks to map the template to the target. However, such multiple deformation method can only learn coarse shape features. It cannot recover complex shape details, which gives the worst geometric similarity between the reconstruction and the ground truth. Ours-un method only makes use of observed visual data, which is at the risk of overfitting to sparse observation. Our NDM method deals with shape reconstruction in a coarse-to-fine fashion and can learn the implicit shape correspondence from a shape manifold. Therefore, it can accurately reconstruct the bi-ventricular shape and generate high quality meshes.

4.5.2 Shape Registration Performance

In Table 2, we show the mean quantitative results of shape registration from ED phase to ES phase. More detailed
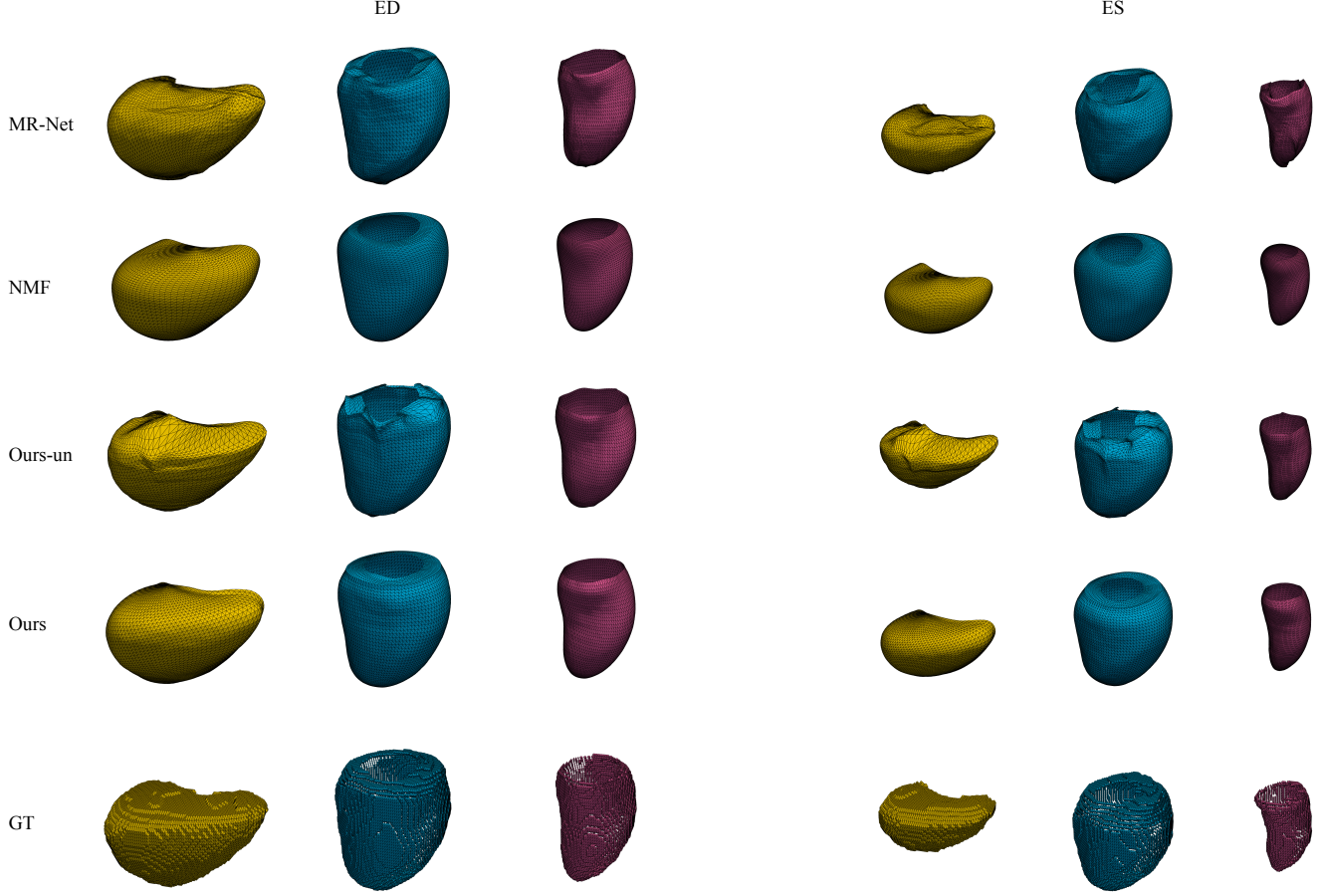
Figure 6. Bi-ventricle shape reconstruction results. For each method, we show results of both the ED and ES phases, in which the LV endo-cardial, epi-cardial and RV surfaces are shown in red, blue and yellow, respectively.
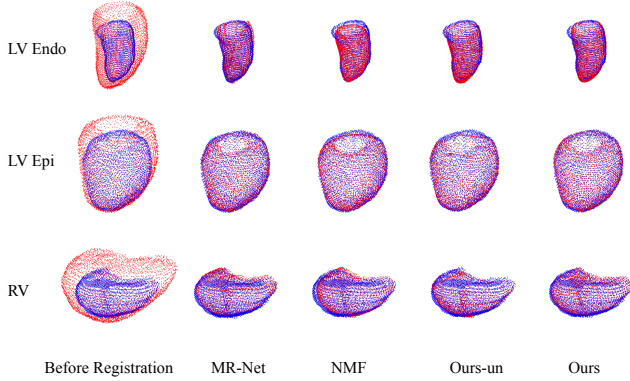

Figure 7. Bi-ventricle shape registration results. For each method, we show registration result from the ED phase to the ES phase, in which the (warped) ED and ES phases are shown in red and blue, respectively.

quantitative results of LV-endo, LV-epi and RV surface registration are presented in Supplementary Material. We

also show an example in Fig. 7. The shape registration is achieved via the learned dense shape correspondence. Thus, the shape registration accuracy reflects the accuracy of implicitly learned dense correspondence. Our method outperforms all baseline methods by a large margin. The results indicate that our method has the potential of achieving accurate heart motion tracking. Note that our method learns 3D dense correspondence between heart shape instances from information extracted from 2D image stacks. This is of great importance for the 3D heart motion field estimation for 2D-based cardiac MR imaging.

4.5.3 Intuitive Interpretation of NDM Parameters

NDM is not only powerful for shape reconstruction and registration, but also enables intuitive interpretation of the heart geometry and deformation parameters, which is a distinct property not shared by any baseline methods. In Fig. 8, we show the five different kinds of parameters of the left ventricle (LV) endo- and epi-cardial surfaces calculated on the test dataset. We can obtain intuitive information from
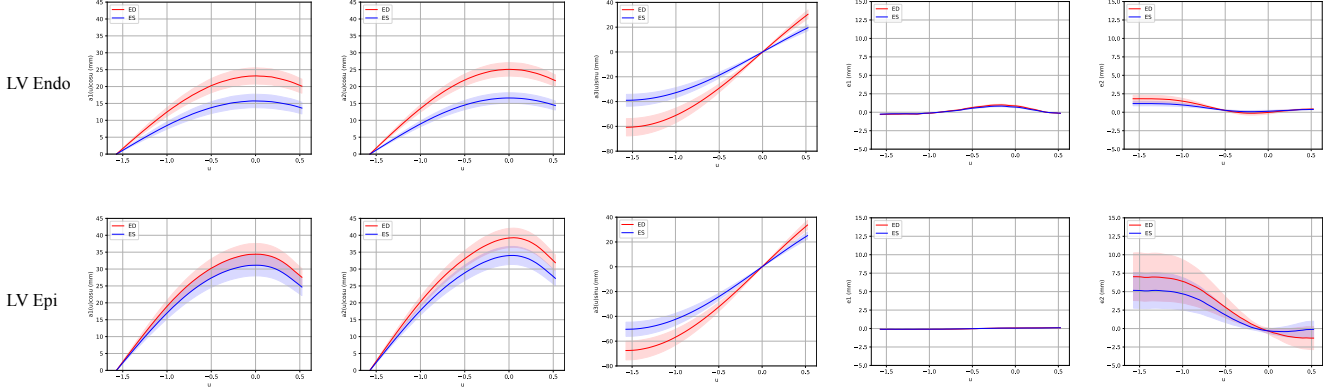
Figure 8. Learned left ventricle geometric and global long axis off set parameters by NDM. Curves show the mean values; color bands show the standard deviations.
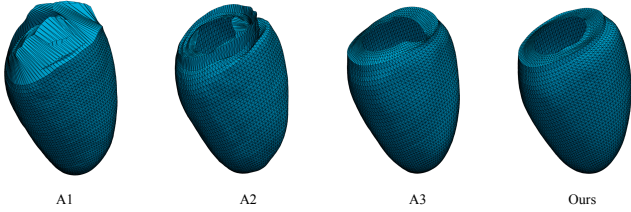

Figure 9. Ablation study results of the LV-epi surface reconstruction. Note the shape reconstruction differences from different models near the basal region.

Model	$\mathcal{L}_d$	$\mathcal{L}_s$	CD $(mm) \downarrow$	SI $\downarrow$
A1	✗	✗	4.49	0.049
A2	✓	✗	3.15	0.049
A3	✗	✓	2.88	0.001
Ours	✓	✓	**2.73**	**0**

Table 3. Ablation of the local deformation regularization terms in NDM on ED phase data.

these plots. The aspect ratios $a'_1(u) = a_1(u)\cos u$ and $a'_2(u) = a_2(u)\cos u$ describe the short axis LV "radius" distribution range along the long axis of the heart. The aspect ratio $a'_3(u) = a_3(u)\sin u$ describes the long axis LV "radius" distribution range along the long axis of the heart. The axis offsets $e_1(u)$ $(e_{xo}(u))$ and $e_2(u)$ $(e_{yo}(u))$ describe the long axis bending information. Importantly, all of these parameters can be directly used by clinicians without any complex post-processing. We can also derive other useful information about the LV from these basic parameters, based on multiple temporal data points. For example, by comparing a'_1, a'_2, a'_3 for ED phase and ES phase, we get an LV contraction metric. By studying the metrics based on populations, we can immediately build an LV myocardium wall motion atlas.

4.5.4 Ablation Study

In Table 3, we present the effects of local deformation regularization terms proposed in our NDM model. We also show an example of LV-epi surface reconstruction in Fig. 9. Our method is different from MR-Net, which uses graph convolutional networks and mesh-based metrics for the deformation regularization. We use NODE to learn a diffeomorphism point flow. The advantage of this method is that we can get rid of complex geometric computation in order to learn plausible shape reconstructions. However, as shown by model A1, A2 and A3, we still need explicit regularization to achieve the goal. The reconstruction differences of these models are manifested mainly around the basal region, where distinct shape variations are observed. Without the explicit local deformation regularization terms, even the use of NODE could lead to unpleasant shape reconstruction overfitting. However, based on our novel design of the NDM model, we can use simple local deformation total amount ($\mathcal{L}_d$) and neighbouring smoothness ($\mathcal{L}_s$) regularization to achieve accurate shape reconstructions.

5. Conclusion

In this work, we proposed a novel neural deformable model to reconstruct bi-ventricular shape from sparse visual data. We inherited a conventional parameter function-based deformable model and incorporated neural ordinary differential equation blocks for local diffeomorphic deformations. We designed a coarse-to-fine learning paradigm to make NDM successfully learn from a shape distribution manifold. Based on such a learning paradigm, only using a simple local deformation regularization could overcome potential overfitting problems. Our NDM model can not only reconstruct bi-ventricular shape accurately, but also enable accurate 3D shape registration. More importantly, our NDM models provide clinicians with intuitive heart shape and deformation parameters in a straightforward manner.

References

[1] Vincent Arsigny, Olivier Commowick, Xavier Pennec, and Nicholas Ayache. A log-euclidean framework for statistics on diffeomorphisms. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 924–931. Springer, 2006. 2

[2] Wenjia Bai, Wenzhe Shi, Antonio de Marvao, Timothy JW Dawes, Declan P O'Regan, Stuart A Cook, and Daniel Rueckert. A bi-ventricular cardiac atlas built from 1000+ high resolution mr images of healthy subjects and an analysis of shape and motion. *Medical image analysis*, 26(1):133–145, 2015. 5

[3] M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61(2):139–157, 2005. 2

[4] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987. 3

[5] Haim Brezis and Haim Brézis. *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer, 2011. 4

[6] Ricky TQ Chen, Brandon Amos, and Maximilian Nickel. Learning neural event functions for ordinary differential equations. *arXiv preprint arXiv:2011.03902*, 2020. 2

[7] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018. 2, 4

[8] Xiang Chen, Nishant Ravikumar, Yan Xia, Rahman Attar, Andres Diaz-Pinto, Stefan K Piechnik, Stefan Neubauer, Steffen E Petersen, and Alejandro F Frangi. Shape registration with learned deformations for 3d shape reconstruction from sparse and incomplete point clouds. *Medical Image Analysis*, 74:102228, 2021. 1, 2, 6

[9] Antonio de Marvao, Timothy JW Dawes, Wenzhe Shi, Christopher Minas, Niall G Keenan, Tamara Diamond, Giuliana Durighel, Giovanni Montana, Daniel Rueckert, Stuart A Cook, et al. Population-based studies of myocardial hypertrophy: high resolution cardiovascular magnetic resonance atlases improve statistical power. *Journal of cardiovascular magnetic resonance*, 16(1):1–10, 2014. 5

[10] Douglas DeCarlo and Dimitris Metaxas. Shape evolution with structural and topological changes using blending. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1186–1205, 1998. 2, 3

[11] Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. *Advances in Neural Information Processing Systems*, 32, 2019. 4

[12] Paul Dupuis, Ulf Grenander, and Michael I Miller. Variational problems on flows of diffeomorphisms for image matching. *Quarterly of applied mathematics*, pages 587–600, 1998. 4

[13] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017. 4

[14] Xi-Sheng Fang et al. The extruded generalized cylinder: A deformable model for object recovery. In *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 174–181. IEEE, 1994. 3

[15] Kunal Gupta. *Neural mesh flow: 3d manifold mesh generation via diffeomorphic flows*. University of California, San Diego, 2020. 2, 4, 6

[16] Viatcheslav Gurev, Ted Lee, Jason Constantino, Hermenegild Arevalo, and Natalia A Trayanova. Models of cardiac electromechanics based on individual hearts imaging data: image-based electromechanical models of the heart. *Biomechanics and modeling in mechanobiology*, 10:295–306, 2011. 1

[17] Idith Haber, Dimitris N Metaxas, and Leon Axel. Three-dimensional motion reconstruction and analysis of the right ventricle using tagged mri. *Medical image analysis*, 4(4):335–355, 2000. 1

[18] Song Han, Dmitry B Goldgof, and Kevin W Bowyer. Using hyperquadrics for shape recovery from range data. In *1993 (4th) International Conference on Computer Vision*, pages 492–496. IEEE, 1993. 3

[19] Zhenhua Hu, Dimitris Metaxas, and Leon Axel. In vivo strain and stress estimation of the heart left and right ventricles from mri images. *Medical image analysis*, 7(4):435–444, 2003. 1

[20] Thomas Joyce, Stefano Buoso, Christian T Stoeck, and Sebastian Kozerke. Rapid inference of personalised left-ventricular meshes by deformation-based differentiable mesh voxelization. *Medical Image Analysis*, 79:102445, 2022. 1

[21] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988. 2

[22] Ruihui Li, Xianzhi Li, Pheng-Ann Heng, and Chi-Wing Fu. Point cloud upsampling via disentangled refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 344–353, 2021. 6

[23] Guo Luo and Pheng-Ann Heng. Lv shape and motion: B-spline-based deformable model and sequential motion decomposition. *IEEE Transactions on Information Technology in Biomedicine*, 9(3):430–446, 2005. 1, 2

[24] PM Nielsen, IJ Le Grice, BH Smaill, and PJ Hunter. Mathematical model of geometry and fibrous structure of the heart. *American Journal of Physiology-Heart and Circulatory Physiology*, 260(4):H1365–H1378, 1991. 1, 2

[25] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4d reconstruction by learning particle dynamics. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5379–5389, 2019. 2

[26] Jinah Park, Dimitris Metaxas, and Leon Axel. Analysis of left ventricular wall motion based on volumetric deformable models and mri-spamm. *Center for Human Modeling and Simulation*, page 99, 1996. 1, 2

[27] Jinah Park, Dimitri Metaxas, Alistair A Young, and Leon Axel. Deformable models with parameter functions for cardiac motion analysis from tagged mri data. *IEEE Transactions on Medical Imaging*, 15(3):278–289, 1996. 2, 3

9

[28] Kyoungju Park, Dimitris N Metaxas, and Leon Axel. Lv-rv shape modeling based on a blended parameterized model. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2002: 5th International Conference Tokyo, Japan, September 25–28, 2002 Proceedings, Part I 5*, pages 753–761. Springer, 2002. 1

[29] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 5

[30] Ashwin Raju, Shun Miao, Dakai Jin, Le Lu, Junzhou Huang, and Adam P Harrison. Deep implicit statistical shape models for 3d medical image delineation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2135–2143, 2022. 5

[31] Lukasz Romaszko, Agnieszka Borowska, Alan Lazarus, David Dalton, Colin Berry, Xiaoyu Luo, Dirk Husmeier, and Hao Gao. Neural network-based left ventricle geometry prediction from cmr images with application in biomechanics. *Artificial Intelligence in Medicine*, 119:102140, 2021. 1

[32] Avan Suinesiaputra, Pierre Ablin, Xenia Alba, Martino Alessandrini, Jack Allen, Wenjia Bai, Serkan Cimen, Peter Claes, Brett R Cowan, Jan D'hooge, et al. Statistical shape modeling of the left ventricle: myocardial infarct classification challenge. *IEEE journal of biomedical and health informatics*, 22(2):503–515, 2017. 1

[33] Shanlin Sun, Kun Han, Deying Kong, Hao Tang, Xiangyi Yan, and Xiaohui Xie. Topology-preserving shape reconstruction and registration via neural diffeomorphic flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20845–20855, 2022. 2, 6

[34] Demetri Terzopoulos and Kurt Fleischer. Deformable models. *The visual computer*, 4(6):306–331, 1988. 2

[35] Demetri Terzopoulos and Dimitri Metaxas. Dynamic 3 d models with local and global deformations: deformable superquadrics. *IEEE Transactions on pattern analysis and machine intelligence*, 13(7):703–714, 1991. 2, 3

[36] Demetri Terzopoulos, Andrew Witkin, and Michael Kass. Constraints on deformable models: Recovering 3d shape and nonrigid motion. *Artificial intelligence*, 36(1):91–123, 1988. 2

[37] Eranga Ukwatta, Hermenegild Arevalo, Martin Rajchl, James White, Farhad Pashakhanloo, Adityo Prakosa, Daniel A Herzka, Elliot McVeigh, Albert C Lardo, Natalia A Trayanova, et al. Image-based reconstruction of three-dimensional myocardial infarct geometry for patient-specific modeling of cardiac electrophysiology. *Medical physics*, 42(8):4579–4590, 2015. 1

[38] Hans C Van Assen, Mikhail G Danilouchkine, Alejandro F Frangi, Sebastián Ordás, Jos JM Westenberg, Johan HC Reiber, and Boudewijn PF Lelieveldt. Spasm: a 3d-asm for segmentation of sparse and arbitrarily oriented cardiac mri data. *Medical image analysis*, 10(2):286–303, 2006. 1, 2

[39] HM Wang, H Gao, XY Luo, C Berry, BE Griffith, RW Ogden, and TJ Wang. Structure-based finite strain modelling of the human left ventricle in diastole. *International journal for numerical methods in biomedical engineering*, 29(1):83–103, 2013. 1

[40] Alistair A Young, Brett R Cowan, Steven F Thrupp, Warren J Hedley, and Louis J Dell'Italia. Left ventricular mass and volume: fast calculation with guide-point modeling on mr images. *Radiology*, 216(2):597–602, 2000. 1, 2

[41] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 4

[42] Yefeng Zheng, Adrian Barbu, Bogdan Georgescu, Michael Scheuering, and Dorin Comaniciu. Fast automatic heart chamber segmentation from 3d ct data using marginal space learning and steerable features. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007. 4, 5

6. Supplementary Material

6.1. Detailed Quantitative Results of Bi-ventricular Shape Reconstruction

In Table 4, we show the detailed Chamfer distance (CD), earth mover's distance (EMD) and point-to-surface distance (P2F) for geometric similarity evaluation of LV-endo, LV-epi and RV shape reconstructions. Our method outperforms all the baseline methods for LV-endo, LV-epi and RV shape reconstruction, respectively.

In Table 5, we show the detailed normal consistency (NC), easy non-manifold face (ENF) ratio and self-intersection (SI) ratio for reconstructed mesh quality evaluation of LV-endo, LV-epi and RV shapes. On average, our method can generate better cardiac meshes than all the baseline methods.

6.2. Detailed Quantitative Results of Bi-ventricular Shape Registration

In Table 6, we show the detailed CD, EMD, P2F for LV-endo, LV-epi and RV shape registration accuracy evaluations. Note that we register shapes from ED phase to ES phase. Our method outperforms all the baseline methods for LV-endo, LV-epi and RV shape registration, respectively.

Phase	Method	CD (mm) ↓				EMD (mm) ↓				P2F (mm) ↓			
		Endo	Epi	RV	avg	Endo	Epi	RV	avg	Endo	Epi	RV	avg
ED	MR-Net	2.56 ± 1.14	3.32 ± 1.10	3.39 ± 1.07	3.09 ± 1.06	6.04 ± 1.19	6.67 ± 1.71	5.44 ± 1.67	6.05 ± 1.17	1.26 ± 0.984	1.32 ± 0.817	1.82 ± 0.945	1.47 ± 0.877
	NMF	5.85 ± 4.43	5.67 ± 4.33	7.40 ± 8.96	6.31 ± 5.54	6.16 ± 1.37	6.61 ± 1.68	6.73 ± 2.27	6.50 ± 1.43	4.07 ± 3.70	3.45 ± 3.88	5.69 ± 8.03	4.40 ± 4.90
	Ours-un	2.72 ± 0.915	4.38 ± 0.978	8.38 ± 4.46	5.16 ± 1.70	5.93 ± 1.31	6.42 ± 1.60	6.24 ± 1.99	6.20 ± 1.26	1.37 ± 0.704	2.15 ± 0.682	5.81 ± 3.56	3.11 ± 1.30
	Ours	**2.25** ± 1.03	**2.91** ± 0.762	**3.02** ± 0.892	**2.73** ± 0.795	**5.70** ± 1.14	**6.38** ± 1.72	**5.33** ± 1.69	**5.80** ± 1.17	**1.01** ± 0.850	**1.06** ± 0.515	**1.43** ± 0.712	**1.17** ± 0.606
ES	MR-Net	2.11 ± 0.898	2.73 ± 0.755	2.28 ± 0.805	2.37 ± 0.729	2.59 ± 0.524	5.27 ± 1.53	5.64 ± 1.63	4.50 ± 0.666	1.18 ± 0.801	1.16 ± 0.539	1.35 ± 0.767	1.23 ± 0.641
	NMF	3.41 ± 3.10	3.74 ± 1.47	4.05 ± 2.83	3.74 ± 2.26	2.76 ± 0.870	5.29 ± 1.38	6.17 ± 1.72	4.74 ± 0.796	2.25 ± 2.35	2.04 ± 1.13	3.11 ± 2.59	2.46 ± 1.86
	Ours-un	1.97 ± 0.631	3.44 ± 0.663	3.27 ± 1.55	2.90 ± 0.732	2.93 ± 0.633	5.14 ± 1.27	5.37 ± 1.56	4.48 ± 0.656	1.15 ± 0.483	1.77 ± 0.433	1.95 ± 1.13	1.63 ± 0.516
	Ours	**1.50** ± 0.542	**2.32** ± 0.582	**1.92** ± 0.823	**1.91** ± 0.556	**2.23** ± 0.481	**4.92** ± 1.42	**5.06** ± 1.49	**4.07** ± 0.607	**0.759** ± 0.397	**0.914** ± 0.356	**0.947** ± 0.605	**0.873** ± 0.392

Table 4. Detailed Chamfer distance (CD), earth mover's distance (EMD) and point-to-surface distance (P2F) for geometric similarity evaluation of LV-endo, LV-epi and RV shape reconstruction, respectively.

Phase	Method	NC ↑				ENF ↓				SI ↓			
		Endo	Epi	RV	avg	Endo	Epi	RV	avg	Endo	Epi	RV	avg
ED	MR-Net	0.713 ± 0.034	0.707 ± 0.033	0.866 ± 0.012	0.762 ± 0.024	1.48 ± 0.004	1.48 ± 0.002	1.47 ± 0.004	1.48 ± 0.003	0.035 ± 0.024	0.016 ± 0.012	0.055 ± 0.018	0.035 ± 0.013
	NMF	0.715 ± 0.031	0.707 ± 0.033	0.854 ± 0.024	0.759 ± 0.024	1.47 ± 0	1.47 ± 0	1.44 ± 0	1.46 ± 0	0.011 ± 0.027	0.016 ± 0.026	0.030 ± 0.046	0.019 ± 0.020
	Ours-un	**0.717** ± 0.033	0.701 ± 0.032	0.835 ± 0.016	0.751 ± 0.022	1.47 ± 0	1.47 ± 0.002	**1.42** ± 0.004	**1.45** ± 0.002	0 ± 0	0 ± 0.001	0.007 ± 0.011	0.002 ± 0.004
	Ours	0.716 ± 0.032	**0.709** ± 0.034	**0.871** ± 0.011	**0.765** ± 0.023	**1.47** ± 0	**1.47** ± 0	1.44±0	1.46 ± 0	**0** ± 0	**0** ± 0.001	**0** ± 0	**0** ± 0
ES	MR-Net	0.708 ± 0.028	0.698 ± 0.033	0.843 ± 0.017	0.750 ± 0.022	**1.46** ± 0.010	1.49 ± 0.003	1.46 ± 0.006	1.47 ± 0.005	0.113 ± 0.042	0.012 ± 0.014	0.078 ± 0.024	0.068 ± 0.018
	NMF	0.716 ± 0.030	0.699 ± 0.034	0.840 ± 0.021	0.752 ± 0.023	1.47 ± 0	1.47 ± 0	1.44 ± 0	1.46 ± 0	0.012 ± 0.025	0.018 ± 0.025	0.051 ± 0.082	0.027 ± 0.030
	Ours-un	**0.719** ± 0.031	0.693 ± 0.033	0.830 ± 0.015	0.747 ± 0.022	1.47 ± 0	1.47 ± 0.001	**1.43** ± 0.004	**1.45** ± 0.001	0 ± 0	0 ± 0	0.004 ± 0.006	0.002 ± 0.002
	Ours	0.715 ± 0.032	**0.701** ± 0.033	**0.857** ± 0.013	**0.758** ± 0.022	1.47 ± 0	**1.47** ± 0	1.44 ± 0	1.46 ± 0	**0** ± 0	**0** ± 0.004	**0** ± 0.002	**0** ± 0.001

Table 5. Detailed normal consistency (NC), easy non-manifold face (ENF) ratio and self-intersection (SI) ratio for reconstructed mesh quality evaluation of LV-endo, LV-epi and RV shape, respectively.

Method	CD (mm) ↓				EMD (mm) ↓				P2F (mm) ↓			
	Endo	Epi	RV	avg	Endo	Epi	RV	avg	Endo	Epi	RV	avg
MR-Net	2.19 ± 0.754	2.82 ± 0.789	2.33±0.737	2.44 ± 0.678	4.23 ± 1.09	5.37 ± 1.01	4.63 ± 1.11	4.74 ± 0.635	1.21 ± 0.560	1.44 ± 0.595	1.35 ± 0.627	1.33 ± 0.532
NMF	3.39 ± 2.48	3.85 ± 1.36	4.02 ± 2.34	3.76 ± 1.84	4.18 ± 1.08	5.94 ± 1.25	4.63 ± 1.28	4.92 ± 0.909	2.19 ± 1.91	2.26 ± 1.07	2.90 ± 2.19	2.45 ± 1.54
Ours-un	1.99 ± 0.635	3.58 ± 0.682	3.51 ± 1.58	3.03 ± 0.739	3.75 ± 0.715	5.37 ± 0.803	4.12 ± 1.05	4.41 ± 0.545	1.21 ± 0.484	2.15 ± 0.486	2.37 ± 1.42	1.91 ± 0.603
Ours	**1.54** ± 0.535	**2.36** ± 0.584	**1.91** ± 0.714	**1.94** ± 0.525	**3.43** ± 0.653	**5.18** ± 0.820	**3.23** ± 0.872	**3.95** ± 0.514	**0.827** ± 0.389	**1.15** ± 0.390	**1.03** ± 0.621	**1.00** ± 0.402

Table 6. Detailed Chamfer distance (CD), earth mover's distance (EMD) and point-to-surface distance (P2F) for shape registration accuracy evaluation of LV-endo, LV-epi and RV, respectively. We register shapes from ED phase to ES phase.