DeepMesh: Mesh-based Cardiac Motion Tracking using Deep Learning

Qingjie Meng,
The Biomedical Image Analysis Group, Department of Computing, Imperial College London, SW7 2AZ, UK

Wenjia Bai,
The Biomedical Image Analysis Group, Department of Computing, Imperial College London, SW7 2AZ, UK; Department of Brain Sciences, Imperial College London

Declan P O'Regan,
The MRC London Institute of Medical Sciences, Imperial College London, W12 0HS, UK

Daniel Rueckert [Fellow, IEEE]
The Biomedical Image Analysis Group, Department of Computing, Imperial College London, SW7 2AZ, UK; Klinikum rechts der Isar, Technical University Munich, Germany

Declan P O'Regan: declan.oregan@imperial.ac.uk

Abstract

3D motion estimation from cine cardiac magnetic resonance (CMR) images is important for the assessment of cardiac function and the diagnosis of cardiovascular diseases. Current state-of-the art methods focus on estimating dense pixel-/voxel-wise motion fields in image space, which ignores the fact that motion estimation is only relevant and useful within the anatomical objects of interest, *e.g.*, the heart. In this work, we model the heart as a 3D mesh consisting of epi- and endocardial surfaces. We propose a novel learning framework, *DeepMesh*, which propagates a template heart mesh to a subject space and estimates the 3D motion of the heart mesh from CMR images for individual subjects. In *DeepMesh*, the heart mesh of the end-diastolic frame of an individual subject is first reconstructed from the template mesh. Mesh-based 3D motion fields with respect to the end-diastolic frame are then estimated from 2D short- and long-axis CMR images. By developing a differentiable mesh-to-image rasterizer, *DeepMesh* is able to leverage 2D shape information from multiple anatomical views for 3D mesh reconstruction and mesh motion estimation. The proposed method estimates vertex-wise displacement and thus maintains vertex correspondences between time frames, which is important for the quantitative assessment of cardiac function across different subjects and populations. We evaluate *DeepMesh* on CMR images acquired from the UK Biobank. We focus on 3D motion estimation of the left ventricle in this work. Experimental results show that the proposed method quantitatively and qualitatively outperforms other image-based and mesh-based cardiac motion tracking methods.

Index Terms

3D motion tracking; 3D mesh reconstruction; cine CMR; deep learning

I Introduction

ESTIMATING left ventricular (LV) myocardial motion is important for the detection of LV dysfunction and the diagnosis of myocardial diseases [1], [2]. Recent works utilize 3D surface meshes to represent the anatomy and assess the ventricular structure and function from meshes, *e.g.*, quantifying pathological cardiac remodeling [3] or characterizing LV motion phenotypes [4]. However, it remains a challenging problem to estimate cardiac motion on meshes directly from images, in particular, to keep the same mesh structure and vertex correspondence. Most recent cardiac motion tracking approaches utilize cine CMR images to estimate a dense motion field which represents pixel-/voxel-wise deformation in the image space, *e.g.*, [2], [5]–[12]. Mapping the deformation from a pixel-/voxel-wise representation to a vertex-wise representation on a cardiac mesh is typically inefficient and can reduce the accuracy of motion estimation. Specifically, a 2D pixel-wise motion field only considers the motion of the heart within a single view plane and does not provide complete 3D motion information. Using post-processing steps to convert 3D voxel-wise motion fields to 3D vertex-wise displacements may impair motion estimation accuracy due to interpolation.

In this work, we propose a novel learning-based method *DeepMesh* for estimating 3D cardiac motion on the heart mesh from 2D cine CMR images. The proposed method propagates a single template mesh to individual subjects and estimates both in-plane and through-plane motion on meshes by integrating information from short-axis (SAX) and long-axis (LAX) view images. Specifically, DeepMesh first utilizes a template heart mesh containing the epi- and endocardial surfaces to reconstruct the mesh at the end-diastolic (ED) frame for an individual heart from the input ED frame multiview images. By deforming this template mesh, the proposed approach maintains the same mesh structure at the ED frame for all subjects. Subsequently, the multi-view images at the ED and t-th frames are utilized to directly estimate the 3D motion on the mesh. The estimated mesh motion explicitly shows the 3D displacement of each vertex from the ED frame to the t-th frame, and thus is able to maintain mesh structure and vertex correspondences between time frames. A differentiable mesh-to-image rasterizer is introduced during training to generate 2D soft segmentations from the 3D mesh. By comparing predicted 2D soft segmentations with ground truth 2D segmentations, the differentiable rasterizer allows leveraging of 2D multi-view anatomical shape information for both 3D mesh reconstruction and motion estimation. During inference, our model generates a sequence of meshes, which characterises the heart motion across the cardiac cycle. Here, in this work, we model the left ventricle as a 3D mesh consisting of epi- and endocardial surfaces and estimate the LV myocardial motion.

Contributions

This paper extends a preliminary version of the work presented at the MICCAI 2022 conference [13]. In addition to the work in [13], the main contributions in terms of methodology and evaluation are summarized as follows:

- We additionally introduce a template-based mesh reconstruction module. This module reconstructs the ED frame mesh of individual subjects from a cardiac template and therefore, enables subsequent mesh-based motion tracking. With the mesh derived from the template, the proposed method is able to maintain the number of vertices and faces in the cohort.

- We add a new regularization term to the motion estimation module in [13] and demonstrate that this leads to an improved performance in motion tracking.

- We conduct a more thorough experimental analysis of the proposed method. We quantitatively and qualitatively evaluate the performance of mesh reconstruction and mesh-based motion tracking. We additionally compare the proposed method with two state-of-the-art motion tracking methods which use multi-view images [12], [13]. Moreover, we perform an extensive ablation study with respect to anatomical views, loss combinations and hyper-parameter selections.

II Related work

A Image-based motion estimation

Many cardiac motion estimation methods, including conventional methods and deep learning-based methods, consider motion tracking within image space. They typically use image registration algorithms to estimate 2D pixel-wise or 3D voxel-wise motion fields.

1 Conventional methods—Image registration has been applied to cardiac motion estimation in previous works. For example, the free form deformation (FFD) method for non-rigid image registration [14] has been widely used for cardiac motion estimation in many recent works, *e.g.*, [2], [6], [8], [15]–[19]. De Craene et al. [20] introduced continuous spatiotemporal B-spline kernels for computing a 4D velocity field, which enforced temporal consistency in motion recovery. Thirion [21] developed the demons algorithm which utilizes diffusing models for image matching and further used it for cardiac motion tracking. Based on this work, Vercauteren et al. [22] introduced a non-parametric diffeomorphic image registration method which has been used for cardiac motion tracking [7].

2 Deep learning-based methods—In recent years, deep convolutional neural networks (CNNs) have inspired the exploration of deep learning-based cardiac motion estimation approaches [23]. Qin et al. [5] proposed a joint deep learning network for simultaneous cardiac segmentation and motion estimation. Their method contains a shared feature encoder which enables a weakly-supervised segmentation. The U-Net architecture [24] has been widely used for learning-based image registration [25], [26] and further for cardiac motion estimation. For example, Zheng et al. [27] proposed a method for cardiac pathology classification based on cardiac motion. This method utilizes a modified U-Net to generate flow maps between the ED frame and any other frame. Balakrishnan et al. [25]

used 3D U-Net to build VoxelMorph for learning-based deformable image registration. Their registration method has been utilized in other cardiac motion tracking works, *e.g.*, [28]. Different from most of these previous deep learning-based methods that aim at 2D motion tracking by only using SAX stacks, Meng et al. [12] focused on 3D motion tracking by fully combining multiple anatomical views. Here, a deep learning model is proposed that learns 3D motion fields from a set of 2D SAX and LAX cine CMR images, which is able to estimate both in-plane and through-plane myocardial motion. Regarding cardiac motion tracking in multiple datasets, Yu et al. [9] considered the distribution mismatch problem and proposed a meta-learning-based online model adaption framework. Towards motion tracking in tagged MRI image, Ye et al. [10] proposed a deep learning model where the motion fields between any two consecutive frames are first computed, and then combined to estimate the Lagrangian motion field between the ED frame and any other frame. Our method aims at 3D cardiac motion tracking from 2D images of multiple anatomical views. In contrast to [12] which estimates 3D motion in image space, our method focuses on estimating 3D motion in mesh space.

B Mesh-based motion estimation

In contrast to dense motion estimation in image space, several other methods focus on anatomical motion estimation in mesh space [29]. These approaches explore mesh matching or mesh registration to estimate the motion field of the mesh. For example, Papademetris et al. [30] proposed a method that uses a biomechanical modeling and shape-tracking approach to estimate the motion of the myocardial mesh. Pan et al. [31] built a 3D mesh to represent material points inside the left ventricle wall and extended 2D Harmonic phase (HARP) technique [32] to 3D for motion tracking of the mesh through a cardiac cycle. Abdelkhalek et al. [33] built a framework to compute mesh displacements via point clouds alignment. These mesh motion estimation approaches compute mesh motion fields only from dynamic shape information, without considering intensity information from images. In contrast, our method combines image information with the myocardial mesh which contains the epi- and endocardial surfaces of the heart. We estimate 3D motion fields on meshes by using the intensity information of 2D images from multiple anatomical views.

C Mesh reconstruction

In practice, the 3D mesh of the heart is not always available. Reconstructing a 3D mesh from images has been well investigated in the literature of general computer vision. Conventional approaches are based on multi-view geometry [34]. Although they can obtain high-quality reconstruction, these approaches are limited by the coverage provided by the multiple views. More recently, deep learning-based approaches are the major trend of 3D shape generation and they can reconstruct 3D meshes from only single or few images. Because of the difficulty of directly generating a feasible mesh structure, most learning-based methods learn shape priors from data and deform a sphere mesh to the target surface, *e.g.*, [35]–[41].

In medical imaging, 3D shape reconstruction of the heart has been studied in the literature. For example, Villard et al. [42] proposed a data fitting method for cardiac surface reconstruction from 2D cardiac contours. This method iteratively optimizes the surface smoothness term and the contour matching term to obtain the 3D mesh of the heart.

However, this method obtains meshes without maintaining vertex correspondences across the cohort. Bello et al. [6] extracted heart surface meshes from image segmentations using the marching cube algorithm. This method also does not maintain vertex correspondences. Romaszko et al. [43] proposed a deep neural network to predict point clouds of heart from images. Following their work, Joyce et al. [44] proposed a mesh fitting method which iteratively optimizes shape parameters (*e.g.*, scalars, orientations) in order to match a mesh to the input 2D segmentations. Xia et al. [45] proposed a method that uses CNNs for statistical shape modeling, in particular, adding phenotypic and demographic information for shape reconstruction. Their method estimates shape parameters and transformation parameters to deform the mean shape of the population for each subject. However, their method needs conventional registration algorithms to generate reference 3D shape information for model training, *i.e.*, reference shape parameters and reference transformation parameters. Different from these previous works, we build a deep neural network that directly predicts the 3D surface mesh of the heart at the ED frame by deforming a cardiac template according to the input 2D multi-view cine CMR images. Our method is able to reconstruct corresponding heart meshes across different subjects, i.e. with a consistent number of vertices and faces.

III   Method

Give a set of CMR images, our goal is to propagate a single template mesh to all subjects and for individual subjects to estimate the heart motion on meshes across the cardiac cycle. Our task is formulated as follows: Let $\{V_{tpl}, F\}$ denote the template mesh, $\{I_0^{sa}, I_0^{2ch}, I_0^{4ch}\}$ denote the 2D SAX, LAX 2-chamber (2CH) and LAX 4-chamber (4CH) view images of the heart at the ED frame and $\{I_t^{sa}, I_t^{2ch}, I_t^{4ch}\}$ denote the multi-view images at the t-th frame. V_{tpl} and F refer to the vertices and faces of the template mesh. T is the number of frames in the cardiac cycle and $0 \leq t \leq T - 1$. We want to reconstruct the 3D heart mesh of individual subjects at the ED frame ($\{\widehat{V}_0, F\}$) from the template, and then, for individual subjects, to estimate a 3D mesh motion field $V_{0 \rightarrow t}$ between the ED and t-th frame by using the corresponding multi-view images. Here, $V_{0 \rightarrow t}$ represents the motion of each vertex from the ED frame to the t-th frame, $\{V_{tpl}, \widehat{V}_0, \Delta V_{0 \rightarrow t}\} \in \mathbb{R}^{N \times 3}$ and N is the number of vertices.

The schematic architecture of the proposed method is shown in Fig. 1. The proposed method can be separated into two main components: First, a mesh reconstruction module reconstructs the 3D mesh of the heart at the ED frame for individual subjects by deforming the template mesh (shown as the red box in Fig. 1). Second, a mesh motion estimation module learns the motion of a myocardial mesh from multiview intensity images and deforms the ED frame mesh to the t-th frame based on the learned 3D mesh motion field (shown as the blue box in Fig. 1). During model training, a differentiable mesh-to-image rasterizer is introduced to yield 2D segmentations of the myocardium in the corresponding 2D planes (in the SAX and LAX orientations) by rasterizing the estimated 3D myocardial mesh. This enables using 2D segmentation information to supervise the mesh reconstruction and motion estimation modules.

A   Mesh reconstruction

This module aims to reconstruct the myocardial mesh for individual subjects at the ED frame. In particular, we leverage multi-view input images to learn a displacement $V_{tpl \to 0}$ that deforms the template mesh to the ED frame mesh of individual subjects vertex-by-vertex. Framework shown in Fig. 2.

1   Deformation estimation—We estimate $V_{tpl \to 0}$ from the input multi-view images of the ED frame. Specifically, a deformation network composed of 2D CNN and 3D CNN is introduced to learn an intermediate 3D voxel-wise displacement $\Phi_{tpl \to 0}$ from the 2D input SAX and LAX view images. The diagram of the deformation network architecture is shown in Fig. 3 (a), where 2D convolutional layers learn 2D features from input images, followed by 3D convolutional layers that further learn 3D representations and predict $\Phi_{tpl \to 0}$. Subsequently, a grid sampler is utilized to generate $V_{tpl \to 0}$ from the obtained $\Phi_{tpl \to 0}$. In detail, for each vertex of the input template, its displacement is sampled from $\Phi_{tpl \to 0}$ by using bi-linear interpolation at the coordinates of this vertex. Therefore, $V_{tpl \to 0}$ contains the displacement of each vertex from the template mesh to the ED frame mesh.

We formulate the deformation estimation as follows,

$$\Delta V_{tpl \to 0} = S(H_D(I_0^{sa}, I_0^{2ch}, I_0^{4ch}), V_{tpl}).$$

(1)

Here, $H_D(\cdot)$ is the deformation network, $S(\cdot, \cdot)$ is the grid sampler and $\Phi_{tpl \to 0} = H_D(I_0^{sa}, I_0^{2ch}, I_0^{4ch})$.

2   Reconstructing the ED frame mesh—With the estimated $V_{tpl \to 0}$, the ED frame mesh ($\{\hat{V}_0, F\}$) of individual subject can be reconstructed by deforming the input template ($\{V_{tpl}, F\}$),

$$\hat{V}_0 = V_{tpl} + \Delta V_{tpl \to 0}.$$

(2)

A Laplacian smoothing loss[1] $\mathscr{L}_{smooth}^{tpl \to 0}$ is used to evaluate the smoothness of the reconstructed ED frame mesh. The Laplacian of a vertex $\hat{v}_0^i$ is defined by $L(\hat{v}_0^i)$,

$$L(\hat{v}_0^i) = \frac{1}{|\mathscr{N}_i|} \sum_{j \in \mathscr{N}_i} (\hat{v}_0^i - \hat{v}_0^j).$$

(3)

Here, $\{\hat{v}_0^i, \hat{v}_0^j\}$ are vertices on $\hat{V}_0$ and $\mathscr{N}_i$ is the set of adjacent vertices to $\hat{v}_0^i$.

---

[1]Implemented by pytorch3d.loss.mesh_laplacian_smoothing()

A surface loss $\mathscr{L}_{surf}$ penalizes the similarity between the reconstructed mesh ($\{\widehat{V}_0, F\}$) and the ground truth mesh ($\{V_0, F\}$) of the ED frame. We use the Chamfer distance[2] as the implementation,

$$\mathscr{L}_{surf} = \frac{1}{|\widehat{V}_0|} \sum_{\widehat{v}_0^i \in \widehat{V}_0} \min_{v_0^j \in V_0} \|\widehat{v}_0^i - v_0^j\|_2^2 + \frac{1}{|V_0|} \sum_{v_0^j \in V_0} \min_{\widehat{v}_0^i \in \widehat{V}_0} \|\widehat{v}_0^i - v_0^j\|_2^2.$$

(4)

In addition, we utilize the Huber loss used in [5], [12] as a regularization term to encourage a smooth intermediate Φ_{tpl},

$$\mathscr{L}_{reg}^{tpl \to 0} = \sqrt{\epsilon + \sum_{i=1}^{Q} \|\nabla \Phi_{tpl}(q_i)\|^2}.$$

(5)

Same to [5], [12], ϵ is set to 0.01. q_i is the i-th voxel and Q denotes the number of voxels.

As we aim to learn 3D dense deformation from 2D sparse images for mesh reconstruction, the current losses have difficulty to guarantee accurate performance. To address this problem, we introduce shape constraint from 2D segmentations as an additional regularization. This regularization term $\mathscr{L}_{shape}^{tpl \to 0}$ is described in detail in Sec. III-C.

B   Mesh motion estimation

In this module, we take multi-view images of the ED frame and the t-th frame as input to estimate a vertex-wise 3D mesh motion field $V_{0 \to t}$. Then, we predict the mesh at the t-th frame by deforming the ED frame mesh reconstructed in the previous module using the 3D motion field $V_{0 \to t}$. Fig. 4 shows the overview of this module.

1   Motion estimation—We estimate $V_{0 \to t}$ from the input images via predicting an intermediate voxel-wise 3D motion field $\Phi_{0 \to t}$. In detail, we build a motion network which consists of 2D CNN and 3D CNN to first learn $\Phi_{0 \to t}$. This motion network combines 2D multi-view images at both the ED frame and the t-th frame to estimate the intermediate 3D voxel-wise motion field $\Phi_{0 \to t}$. The diagram of the motion network architecture is in Fig. 3 (b), where 2D convolutional layers learn 2D features from two time frames and 3D convolutional layers predict $\Phi_{0 \to t}$. The obtained $\Phi_{0 \to t}$ represents the motion of image voxels from the ED frame to the t-th frame. Then, a grid sampler is utilized to generate $V_{0 \to t}$ from the obtained $\Phi_{0 \to t}$ based on the vertices on the reconstructed ED frame mesh ($\widehat{V}_0$) and bi-linear interpolation. $V_{0 \to t}$ represents the motion field of each vertex from the ED frame to the t-th frame. Overall, $V_{0 \to t}$ is estimated from the input multi-view images by

$$\Delta V_{0 \to t} = S(H_M(I_0^{sa}, I_0^{2ch}, I_0^{4ch}, I_t^{sa}, I_t^{2ch}, I_t^{4ch}), \widehat{V}_0).$$

---

[2]Implemented by pytorch3d.loss.chamfer_distance()

(6)

Here, $H_M(\cdot, \cdot)$ is the motion network and $\Phi_{0 \to t} = H_M(\cdot)$.

2   Mesh prediction—With the estimated $V_{0 \to t}$, the reconstructed ED frame mesh $(\{\hat{V}_0, F\})$ can be deformed to the t-th frame $(\{\hat{V}_t, F\})$ by

$$\hat{V}_t = \hat{V}_0 + \Delta V_{0 \to t}.$$

(7)

As ground truth mesh displacement is usually unavailable, $V_{0 \to t}$ can not be directly evaluated. Instead, we evaluate $\Phi_{0 \to t}$ in a self-supervised manner. We transform the SAX stack of the t-th frame (I_t^{sa}) to the ED frame using $\Phi_{0 \to t}$ via a spatial transformer network [46]. By minimizing the image similarity loss in Eq. 8, $\Phi_{0 \to t}$ is encouraged to reflect the motion of the myocardium.

$$\mathcal{L}_{sim} = \| I_0^{sa} - I_t^{sa} \circ \Phi_{0 \to t} \|^2$$

(8)

Similar to Eq. 3, the smoothness of the predicted t-th frame mesh is evaluated by a Laplacian smoothing loss[1] $\mathcal{L}_{smooth}^{0 \to t}$.

The gradients of the intermediate $\Phi_{0 \to t}$ is penalized by the Huber loss similar to Eq. 5, $\mathcal{L}_{reg}^{0 \to t} = \sqrt{\epsilon + \sum_{i=1}^{Q} \| \nabla \Phi_{0 \to t}(q_i) \|^2}$.

For mesh motion estimation, we also introduce shape constraint to better learn 3D dense deformation from 2D sparse images. This regularization term ($\mathcal{L}_{shape}^{0 \to t}$) is described in detail in Sec. III-C.

C   Differentiable mesh-to-image rasterizer

As ground truth 3D deformation is usually unavailable, we want to use 2D anatomical shape information to further supervise both 3D mesh reconstruction and motion estimation. To achieve this, we propose a differentiable mesh-to-image rasterizer to extract 2D soft contours of the myocardium from the predicted 3D heart mesh at the ED frame and the t-th frame. By comparing with the ground truth 2D myocardial contours, the differentiable rasterizer enables using sparse 2D shape information from multiple views to supervise 3D mesh reconstruction and motion estimation.

The input of the differentiable rasterizer is the predicted 3D mesh of the myocardium $\{\hat{V}_s, F\}$. The outputs are 2D contours of the myocardium intersecting on SAX, 2CH and 4CH view planes ($\{P_s^{sa}, P_s^{2ch}, P_s^{4ch}\}$). Here $s = \{0, t\}$ refers to the ED frame and the t-th frame, respectively. When extracting a 2D plane from 3D mesh, the vertices on the 3D mesh may not perfectly lie in the 2D plane. Therefore, we compute the probability of vertices lying on the plane, which is important for maintaining the differentiability. Specifically, we use

probability maps to represent the 2D soft contours of the myocardium. Each pixel on the probability map represents the probability of a vertex from the 3D myocaridal mesh lying on a specific 2D plane. The closer a vertex to a plane, the higher probability the vertex lies on the plane. Fig.5 illustrates the rasterizer.

In detail, the coordinates of a vertex $\hat{v}_s^i(\hat{v}_s^i \in \hat{V}_s, i = [0, 1, ..., N])$ are first transformed to the image space of different anatomical planes using the information about the relative position in the DICOM header of 2D images, $e.g.$, $(x_s^{ik}, y_s^{ik}, z_s^{ik})$ is the transformed coordinates of $\hat{v}_s^i$ and k is the target 2D plane. Then, the probability of each vertex being on plane k is estimated according to their distance:

$$p_s^{ik} = e^{-\tau(d_s^{ik})^2}, \quad d_s^{ik} = \left|z_s^{ik} - z^k\right|, \quad k = \{sa, 2ch, 4ch\}$$

(9)

Here p_s^{ik} refers to the probability of $\hat{v}_s^i$ belonging to the plane k and τ is the hyper-parameter which controls the sharpness the exponential function. d_s^{ik} is the distance between $\hat{v}_s^i$ and the plane k, and z^k is the slice corresponding to the plane k. The vertices satisfying $d_s^{ik} < 1$ are selected as the intersection of 3D mesh $\{\hat{V}_s, F\}$ and 2D plane k. The probability values of these vertices form the probability map P_s^k.

The obtained 2D probability maps are compared to 2D ground truth binary segmentations $\{B_s^{sa}, B_s^{2ch}, B_s^{4ch}\}$. Here, only ground truth contours of the myocardium are used and we compare between contours. We utilize a weighted Hausdorff distance[3](WHD(·, ·)) [47] to measure the similarity between these contours. $\mathscr{L}_{shape}^{tpl \to 0}$ is the shape regularization term for the mesh reconstruction module,

$$\mathscr{L}_{shape}^{tpl \to 0} = \sum_{k = \{sa, 2ch, 4ch\}}\mathrm{WHD}(P_0^k, B_0^k).$$

(10)

$\mathscr{L}_{shape}^{0 \to t}$ is the shape regularization term for the mesh motion tracking module. It is the same format to Eq. 10 but the input are P_t^k, B_t^k.

As we use an exponential function (Eq. 9) for the rasterization, when minimizing loss function ($e.g.$, Eq. 10), the gradient can be back-propagated to train the networks. Therefore, the exponential function enables the differentiability of the rasterization, and thus enables end-to-end model training.

D Optimization

Our model is trained by two stages. The first stage is to train the mesh reconstruction module (*i.e.*, Deformation Network $H_D(\cdot)$) by minimizing $\mathscr{L}_{recon}$ (Eq. 11). The inputs are

---

[3]Implemented by https://github.com/javiribera/locating-objects-without-bboxes

the template mesh and the multi-view images at the ED frame. The output is vertex-wise displacement which deforms the template mesh to individual subject.

$$\mathscr{L}_{recon} = \mathscr{L}_{shape}^{tpl \to 0} + \lambda_1 \mathscr{L}_{smooth}^{tpl \to 0} + \beta_1 \mathscr{L}_{surf} + \gamma_1 \mathscr{L}_{reg}^{tpl \to 0}.$$

(11)

The second stage is to train the mesh motion estimation module (*i.e.*, Motion Network $H_M(\cdot)$) by minimizing $\mathscr{L}_{motion}$ (Eq. 12). The inputs are the multi-view images of the ED frame and frame t. The output is mesh motion field. For each training iteration, frame t is randomly selected from the cardiac cycle.

$$\mathscr{L}_{motion} = \mathscr{L}_{shape}^{0 \to t} + \lambda_2 \mathscr{L}_{smooth}^{0 \to t} + \beta_2 \mathscr{L}_{sim} + \gamma_2 \mathscr{L}_{reg}^{0 \to t}.$$

(12)

Here, $\{\lambda_i, \beta_i, \gamma_i\}_{i=\{1,2\}}$ are hyper-parameters chosen experimentally depending on the dataset. We use the Adam optimizer (learning rate = 10^{-4}) to update the network parameters. Our model is implemented by Pytorch and is trained on a NVIDIA RTX A5000 GPU with 24GB of memory.

IV Experiments

We evaluate the performance of 3D mesh reconstruction and mesh motion tracking on LV myocardium. We compare the proposed method, named as DeepMesh, with other image-based and mesh-based cardiac motion tracking methods. We explore the effectiveness of different loss components and the influence of the hyper-parameters. We show the key results in the main paper[4]. The dynamic motion tracking videos can be found in https://github.com/ImperialCollegeLondon/DeepMesh.

A Experiment setups

1 Data—Experiments were performed on randomly selected 530 subjects from the UK Biobank study [48]. Each subject contains SAX, 2CH and 4CH view cine CMR sequences and each sequence contains 50 frames. SAX view images were resampled by linear interpolation from a spacing of $\sim 1.8 \times 1.8 \times 10mm$ to a spacing of $1.25 \times 1.25 \times 2mm$ while 2CH and 4CH view images were resampled from $\sim 1.8 \times 1.8mm$ to $1.25 \times 1.25mm$. Based on the center of the intersecting line between the middle slice of the SAX stack and the LAX view images, the SAX, 2CH and 4CH view images are cropped to cover the whole LV in the center. The input LV template mesh is provided by [49]. This template contains 22, 043 vertices and 43, 840 faces. For model training, 2D segmentations are used to supervise mesh reconstruction and motion tracking. The 2D binary segmentations used in Eq. 10 were extracted from a 3D high resolution segmentation. This 3D high resolution segmentation is generated via an automated tool provided in [50], followed by manual quality control. We use 3D myocardial meshes of the ED frame and the end-systolic (ES) frame for evaluation.

---

[4]Code is at DOI: 10.5281/zenodo.8200635

These ground truth 3D meshes are reconstructed from the 3D high resolution segmentations using the marching cube algorithm. We split the dataset into 400/50/80 for train/validation/ test and train the proposed model for 300 epochs. We choose the hyper-parameters using grid search and select the hyper-parameters with the best performance on the validation data. Specifically, the hyper-parameters in Eq. 11 are chosen from $\lambda_1 = [10, 20, 30, 40, 50]$, $\beta_1 = [0.1, 0.3, 0.5, 0.7, 0.9]$ and $\gamma_1 = [0.1, 0.3, 0.5, 0.7, 0.9]$, and are selected as $\lambda_1 = 20$, $\beta_1 = 0.5$, $\gamma_1 = 0.5$. In Eq. 12, the hyper-parameters are chosen from $\lambda_2 = [100, 130, 150, 170, 190]$, $\beta_2 = [10, 20, 30, 40, 50]$ and $\gamma_2 = [0.1, 0.3, 0.5, 0.7, 0.9]$ and are selected as $\lambda_2 = 150$, $\beta_2 = 20$, $\gamma_2 = 0.5$. In Eq. 9, we select $\tau = 3$ from $\tau = [2, 3]$.

2 Evaluation metrics—For evaluating the performance of 3D motion tracking on meshes, we compared the predicted 3D mesh and the ground truth 3D mesh at the ES frame. In addition, we extract 2D contours of the myocardium at SAX and LAX view planes from the predicted 3D meshes, and then compare the extracted 2D contours with the ground truth 2D contours (extracted from ground truth 3D meshes). The following metrics are used for evaluation: Surface distance, Hausdorff distance (HD) and Boundary F-score (BoundF). The surface distance evaluates the distance between the predicted and the ground truth meshes. The Hausdorff distance and Boundary F-score compare the predicted and the ground truth 2D myocardium contours on SAX, 2CH and 4CH view planes. The Hausdorff distance quantifies the contour distance while Boundary F-score evaluates contour alignment accuracy as described in [51]–[53]. Here, to compute the Hausdorff distance at the SAX view, we average the Hausdorff distance of the second slice (slice 1), the middle slice (slice 4) and the second last slice (slice 7).

3 Baseline methods—We compared the proposed method with five state-of-the-art cardiac motion tracking approaches, including two conventional methods and three learning-based methods. The two conventional methods are a B-spline free form deformation (FFD) algorithm[5] [14] and a diffeomorphic Demons (dDemons) algorithm[6] [22] which have been used in many recent cardiac motion tracking works [2], [6]–[8], [18], [19]. For the learning-based method, the UNet architecture has been used in many recent works for image registration [25], [26], [28], and thus our third baseline is a deep learning method with 3D-UNet[7] [54]. In addition, we compared the proposed method with MulViMotion[8] [12] and MeshMotion [13] which are two deep learning-based methods that utilize multi-view cardiac CMR images for 3D motion tracking. For fair comparison, we evaluated several sets of hyper-parameter values for all methods and selected hyper-parameters that achieve the best Hausdorff distance on the validation set.

B Mesh-based motion tracking

1 Mesh reconstruction performance—The proposed method first reconstructs the mesh of the ED frame for each test subject. Fig. 6 (a) shows that the reconstructed mesh fits the ground truth mesh for a sample case. We extracted SAX, 2CH and 4CH view planes

---

[5]Implemented by using the MIRTK toolkit: http://mirtk.github.io/
[6] https://github.com/InsightSoftwareConsortium/SimpleITK-Notebooks/Python/66_Registration_Demons.ipynb
[7] https://github.com/wolny/pytorch-3dunet
[8] https://github.com/qmeng99/Multiview-Motion-Estimation-for-3D-cardiac-motion-tracking

from the reconstructed ED frame mesh and generated 2D segmentations on different view planes. Fig. 6 (b) and Table I qualitatively and quantitatively show the effectiveness of the mesh reconstruction by comparing the generated and the ground truth 2D myocardial contours.

2   Mesh motion estimation performance—Following mesh reconstruction, the proposed method estimates mesh motion fields in the full cardiac cycle. For each test subject, with the obtained vertex-wise motion fields { $V_t | t = [0, 49]$ }, the reconstructed ED frame mesh is deformed to the t-th frame. Red meshes in Fig. 7 shows that the estimated mesh motion field $V_{0 \to t}$ enables 3D myocardial motion tracking on meshes. In addition, we extracted SAX/2CH/4CH view planes from the predicted t-th frame mesh and generated the predicted 2D myocardium contours on different view planes. Fig. 7 shows the effectiveness of $V_{0 \to t}$ by comparing the predicted and the ground truth 2D myocardium contours.

3   Comparison study—We compare the proposed method with baseline methods for the performance of motion estimation across the cardiac cycle. Fig. 8 demonstrates that MulViMotion [12], MeshMotion [13] and the proposed method are able to estimate both in-plane and through-plane motion while other methods only show motion within SAX plane. This is because [12], [13] and our method take full advantage of both SAX and LAX view images. Different from MulViMotion [12] which estimates a voxel-wise motion field and generates 3D meshes from segmentations, the proposed method directly estimates the motion of each vertex on the heart mesh, and thus is able to keep the number of vertex and the vertex correspondences across the cardiac cycle. In contrast to MeshMotion [13] where the ED frame mesh of an individual heart is needed before motion tracking, the proposed method directly reconstructs the ED frame mesh by propagating from a template mesh. It integrates mesh reconstruction and mesh tracking into a single framework and also ensures the consistency of the meshes across different subjects. In addition, compared to [13], we add a regularization loss $\mathcal{L}_{reg}^{0 \to t}$ in this work to penalize the smoothness of the intermediate dense motion field ($\Phi_{0 \to t}$). The results show that the proposed method achieves smoother LV basal part than [13], *e.g.*, in $t = 20$ and $t = 40$ frame in Fig. 8.

We further compare different methods by estimating the 3D motion field from ED frame to ES frame, which shows the largest deformation. Table II shows the quantitative comparison results and Fig. 9 shows the qualitative results. From Table II, we observe that the proposed method outperforms all baseline methods and achieves the best performance regarding SAX, 2CH and 4CH view segmentations. In addition, the proposed method obtains the ES frame mesh which is most similar to the ground truth ES frame mesh in Fig. 9. These results demonstrate the effectiveness of the proposed method for estimating 3D mesh motion fields.

4   Ablation study—For the proposed method, we explore the effects of using different anatomical views and loss combinations in the mesh reconstruction and the mesh motion estimation. We utilize Hausdorff Distance (HD) for the evaluation. Table III and Table IV show that adding the LAX view images improves the performance. This might be because LAX views can introduce high-resolution through-plane information for 3D motion estimation. These tables also show that proposed method with all the losses performs best in

both mesh reconstruction and motion tracking, which illustrates the importance of each loss component.

5 The influence of hyper-parameters—We evaluate the performance of mesh reconstruction and mesh motion estimation under various values of the hyper-parameters. Specifically, we compute Hausdorff distance (HD) based on the predicted and ground truth 2D myocardium contours on SAX, 2CH and 4CH view planes. We compare the contours of the ED frame for the mesh reconstruction and compare the contours of the ES frame for the mesh motion estimation. Fig. 10 shows that in contrast to LAX views, the performance on the SAX view are not sensitive to hyper-parameters. This might because the SAX stacks contain multiple slices while the 2CH and 4CH view only have a single slice for evaluation. From the last row in Fig. 10 (a), we observe that a weak or a strong regularization on voxel-wise displacement may reduce the accuracy of mesh reconstruction.

V Discussion

In the mesh motion estimation framework presented in this work, we predict the motion field of the heart mesh by sampling from an intermediate voxel-wise 3D motion field. An alternative to our method would be to estimate mesh motion field directly from input images via fully connected layers without intermediate voxel-wise 3D motion estimation. However, using fully connected layers to estimate the displacement of $\sim 20K$ vertices needs large GPU memory, which may not always be available.

We use the weighted Hausdorff Distance to compare the extracted 2D contours and the ground truth 2D contours of myocardium in $\mathcal{L}_{shape}^{tpl \to 0}$ and $\mathcal{L}_{shape}^{0 \to t}$. Other boundary similarity measurements that can evaluate the distance between soft-labeled and hard-labeled point sets may also be applied to this loss component in our task.

When evaluating motion estimation, we quantitatively evaluated the performance on the ES frame. This is because 3D ground truth meshes are only available at the ED and ES frames in our current dataset. More importantly, ES frame has the largest deformation from the ED frame, which is the most challenging case in motion estimation. Besides, using the ES frame for quantitative evaluation is same to other previous works, such as [5], [7], [55].

We separately train the mesh reconstruction module and the mesh motion estimation module during training but the proposed method is end-to-end trainable. The probability map (2D soft contours) obtained from differentiable mesh-to-image rasterizer enables the differentiability of the rasterization. However, simultaneously training mesh reconstruction and mesh motion estimation may increase the complexity of hyper-parameter tuning.

The proposed deep neural network in the mesh reconstruction module focuses on deforming the template mesh to the ED frame mesh of individual subjects. To move the template mesh to individual subject space before mesh reconstruction, we utilize the information about the relative position in the DICOM header of 2D images. Fig. 11 shows an example of moving the template mesh to a subject space during data pre-processing.

Our evaluation has been conducted on LV myocardial motion tracking because it is important for clinical assessment of cardiac function. However, the proposed method is not limited to LV myocardium. Our model can be easily adapted to 3D right ventricular myocardial motion tracking by using the corresponding template mesh and the ground truth 2D contours during training.

Table III and Table IV show that only using shape regularization ($\mathcal{L}_{shape}^{tpl \rightarrow 0}$ and $\mathcal{L}_{shape}^{0 \rightarrow t}$) achieves second best quantitative results. However, Fig. 12 demonstrates that shape regularization alone is insufficient for good qualitative results while other regularization terms make contributions as well, to surface smoothness, accurate deformation and deformation smoothness, accordingly.

The proposed method is trained and evaluated on healthy subjects, where we aim to demonstrate the effectiveness of the methodology. We acknowledge that the current trained model may not achieve best performance on pathological data, especially heart with specific diseases. To address this limitation, one possible solution is to include more pathological cases to the training set and re-train the model. In addition, there can be large deformation between the template mesh and pathological heart, for which we may need to add extra regularization terms to the template-based mesh reconstruction module.

We believe that our mesh-based motion tracking method can benefit a variety of clinical applications. The proposed model can provide an accurate and holistic estimation of 3D geometry and motion, which can be used for clinical association studies. For example, we can model the associations between cardiac motion (either globally, or vertex-wise) with demographics (*e.g.*, age, gender), genetics, diseases, and etc. In particular, as our method maintains the anatomical correspondence of the cardiac meshes (*i.e.*, the number of vertices and faces) in the cohort, it can facilitate learning complex motion features for specific tasks from a population. This can potentially lead to motion-related traits for early diagnosis of diseases or for monitoring disease progression. In addition, our method can support biophysical modeling by using meshes as input for mechanical simulations. This can potentially improve our understanding of cardiac physiology. Also, the predicted sequence of meshes can be used for computing conventional volumetric and functional biomarkers (*e.g.*, ED volume, ejection fraction). The challenge might be to design new computational methods on meshes instead of on segmentations that used in many existing clinical studies.

VI Conclusion

In this paper, we propose a novel deep learning method for template-guided mesh-based cardiac motion tracking. The proposed method reconstructs the 3D heart mesh of the reference frame and estimate per-vertex motion field from 2D SAX and LAX view CMR images. The proposed method enables both mesh reconstruction and mesh motion tracking. It is also capable of maintaining the number of vertices and vertex correspondences across the cardiac cycle. Experimental results demonstrate the effectiveness of the proposed method compared with other competing methods.

Acknowledgments

References

[1]. Claus P, Omar AMS, Pedrizzetti G, Sengupta PP, Nagel E. Tissue tracking technology for assessing cardiac mechanics: Principles, normal values, and clinical applications. JACC Cardiovasc Imaging. 2015; 8 (12) 1444–1460. [PubMed: 26699113]

[2]. Puyol-Antón E, et al. Regional multi-view learning for cardiac motion analysis: Application to identification of dilated cardiomyopathy patients. IEEE Trans Biomed Eng. 2019; 66 (4) 956–966. [PubMed: 30113891]

[3]. Mansi T, et al. A statistical model for quantification and prediction of cardiac remodelling: Application to Tetralogy of Fallot. IEEE Trans Med Imaging. 2011; 30 (9) 1605–1616. [PubMed: 21880565]

[4]. de Marvao A, et al. Precursors of the hypertensive heart phenotype develop in normotensive adults: a high resolution 3D MRI study. JACC Cardiovasc Imaging. 2015.

[5]. Qin, C; , et al. Joint learning of motion estimation and segmentation for cardiac MR image sequences; MICCAI; 2018.

[6]. Bello G, et al. Deep learning cardiac motion analysis for human survival prediction. Nat Mach Intell. 2019; 1: 95–104. [PubMed: 30801055]

[7]. Qin, C; Wang, S; Chen, C; Qiu, H; Bai, W; Rueckert, D. Biomechanics-informed neural networks for myocardial motion tracking in MRI; MICCAI; 2020.

[8]. Bai W, et al. A population-based phenome-wide association study of cardiac and aortic structure and function. Nat Med. 2020; 26: 1654–1662. [PubMed: 32839619]

[9]. Yu, H; , et al. FOAL: Fast online adaptive learning for cardiac motion estimation; CVPR; 2020.

[10]. Ye, M; , et al. Deeptag: An unsupervised deep learning method for motion tracking on cardiac tagging magnetic resonance images; CVPR; 2021.

[11]. Loecher M, Perotti LE, Ennis DB. Using synthetic data generation to train a cardiac motion tag tracking neural network. Med Imag Anal. 2021; 74

[12]. Meng Q, et al. MulViMotion: Shape-aware 3D myocardial motion tracking from multi-view cardiac MRI. IEEE Trans Med Imaging. 2022.

[13]. Meng, Q; Bai, W; Liu, T; O'Regan, DP; Rueckert, D. Mesh-based 3d motion tracking in cardiac mri using deep learning; MICCAI; 2022.

[14]. Rueckert D, Sonoda L, Hayes C, Hill D, Leach M, Hawkes D. Nonrigid registration using free-form deformations: application to breast MR images. IEEE Trans Med Imaging. 1999; 18 (8) 712–721. [PubMed: 10534053]

[15]. Chandrashekara R, Mohiaddin R, Rueckert D. Analysis of 3D myocardial motion in tagged MR images using nonrigid image registration. IEEE Trans Med Imaging. 2004; 23 (10) 1245–1250. [PubMed: 15493692]

[16]. Shen, D; Sundar, H; Xue, Z; Fan, Y; Litt, H. Consistent estimation of cardiac motions by 4D image registration; MICCAI; 2005.

[17]. Shi W, et al. A comprehensive cardiac motion estimation framework using both untagged and 3-D tagged MR images based on nonrigid registration. IEEE Trans Med Imaging. 2012; 31: 1263–1275. [PubMed: 22345530]

[18]. Tobon-Gomez C, et al. Benchmarking framework for myocardial tracking and deformation algorithms: An open access database. Med Imag Anal. 2013; 17 (6) 632–648.

[19]. Puyol-Antón, E; , et al. Fully automated myocardial strain estimation from cine MRI using convolutional neural networks; ISBI; 2018.

[20]. Craene MD, et al. Temporal diffeomorphic free-form deformation: Application to motion and strain estimation from 3D echocardiography. Med Imag Anal. 2012; 16 (2) 427–450.

[21]. Thirion J-P. Image matching as a diffusion process: an analogy with Maxwell's demons. Med Imag Anal. 1998; 2 (3) 243–260.

[22]. Vercauteren, T; Pennec, X; Perchant, A; Ayache, N. Non-parametric diffeomorphic image registration with the demons algorithm; MICCAI; 2007.

[23]. Duchateau N, King A, De Craene M. Machine learning approaches for myocardial motion and deformation analysis. Frontiers in Cardiovascular Medicine. 2020; 6

[24]. Ronneberger, O; Fischer, P; Brox, T. U-Net: Convolutional networks for biomedical image segmentation; MICCAI; 2015. 234–241.

[25]. Balakrishnan G, Zhao A, Sabuncu MR, Guttag JV, Dalca AV. Voxelmorph: A learning framework for deformable medical image registration. IEEE Trans Med Imaging. 2019; 38 (8) 1788–1800.

[26]. Xu, Z; , et al. Adversarial uni-and multi-modal stream networks for multimodal image registration; MICCAI; 2020. 222–232.

[27]. Zheng Q, Delingette H, Ayache N. Explainable cardiac pathology classification on cine MRI with motion characterization by semi-supervised learning of apparent flow. Med Imag Anal. 2019; 56: 80–95.

[28]. Ta, K; Ahn, SS; Stendahl, JC; Sinusas, AJ; Duncan, JS. A semi-supervised joint network for simultaneous left ventricular motion tracking and segmentation in 4D echocardiography; MICCAI; 2020.

[29]. Wang H, Amini AA. Cardiac motion and deformation recovery from mri: A review. IEEE Transactions on Medical Imaging. 2012; 31 (2) 487–503. [PubMed: 21997253]

[30]. Papademetris X, Sinusas AJ, Dione DP, Duncan JS. Estimation of 3D left ventricular deformation from echocardiography. Med Imag Anal. 2001; 5 (1) 17–28.

[31]. Pan L, Prince J, Lima J, Osman N. Fast tracking of cardiac motion using 3d-harp. IEEE Transactions on Biomedical Engineering. 2005; 52 (8) 1425–1435. [PubMed: 16119238]

[32]. Osman N, McVeigh E, Prince J. Imaging heart motion using harmonic phase MRI. IEEE Trans Biomed Eng. 2000; 19 (3) 186–202.

[33]. Abdelkhalek M, Aguib H, Moustafa M, Elkhodary K. Enhanced 3D myocardial strain estimation from multi-view 2D CMR imaging. arXiv preprint. 2020. arXiv:200912466

[34]. Hartley R, Zisserman A. Multiple View Geometry in Computer Vision. 2003.

[35]. Kanazawa, A; Tulsiani, S; Efros, AA; Malik, J. Learning category-specific mesh reconstruction from image collections; ECCV; 2018.

[36]. Wang, N; Zhang, Y; Li, Z; Fu, Y; Liu, W; Jiang, Y-G. Pixel2mesh: Generating 3d mesh models from single rgb images; ECCV; 2018.

[37]. Pan, J; Han, X; Chen, W; Tang, J; Jia, K. Deep mesh reconstruction from single rgb images via topology modification networks; ICCV; 2019.

[38]. Gupta, K; Chandraker, M. Neural mesh flow: 3d manifold mesh generation via diffeomorphic flows; NeurIPS; 2020.

[39]. Shi, Y; Ni, B; Liu, J; Rong, D; Qian, Y; Zhang, W. Geometric granularity aware pixel-to-mesh; ICCV; 2021.

[40]. Amiranashvili, T; Lüdke, D; Li, H; menze, bjoern; Zachow, S. Learning shape reconstruction from sparse measurements with neural implicit functions; MIDL; 2022.

[41]. Sun, S; Han, K; Kong, D; Tang, H; Yan, X; Xie, X. Topology-preserving shape reconstruction and registration via neural diffeomorphic flow; Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022. 20845–20855.

[42]. Villard B, Grau V, Zacur E. Surface mesh reconstruction from cardiac mri contours. Journal of Imaging. 2018; 4 (1)

[43]. Romaszko L, et al. Neural network-based left ventricle geometry prediction from cmr images with application in biomechanics. Artificial Intelligence in Medicine. 2021; 119 102140 [PubMed: 34531009]

[44]. Joyce T, Buoso S, Stoeck CT, Kozerke S. Rapid inference of personalised left-ventricular meshes by deformation-based differentiable mesh voxelization. Medical Image Analysis. 2022; 79 102445 [PubMed: 35468554]

[45]. Xia Y, et al. Automatic 3d+t four-chamber cmr quantification of the uk biobank: integrating imaging and non-imaging data priors at scale. Medical Image Analysis. 2022; 80 102498 [PubMed: 35665663]

[46]. Jaderberg, M; Simonyan, K; Zisserman, A; Kavukcuoglu, K. Spatial transformer networks; NeurIPS; 2015.

[47]. Ribera, J; Guera, D; Chen, Y; Delp, EJ. Locating objects without bounding boxes; CVPR; 2019. 6472–6482.

[48]. Petersen S, et al. UK Biobank's cardiovascular magnetic resonance protocol. J Cardiovasc Magn Reson. 2015; 18

[49]. Bai W, et al. A bi-ventricular cardiac atlas built from 1000+ high resolution MR images of healthy subjects and an analysis of shape and motion. Medical Image Anal. 2015; 26 (1) 133–145.

[50]. Duan J, et al. Automatic 3D bi-ventricular segmentation of cardiac images by a shape-refined multi-task deep learning approach. IEEE Trans Med Imaging. 2019; 38 (9) 2151–2164. [PubMed: 30676949]

[51]. Perazzi, F; Pont-Tuset, J; McWilliams, B; Van Gool, L; Gross, M; Sorkine-Hornung, A. A benchmark dataset and evaluation methodology for video object segmentation; CVPR; 2016.

[52]. Cheng, D; Liao, R; Fidler, S; Urtasun, R. Darnet: Deep active ray network for building segmentation; CVPR; 2019. 7423–7431.

[53]. Gur, S; Shaharabany, T; Wolf, L. End to end trainable active contours via differentiable rendering; ICLR; 2020.

[54]. Cicek, Ö; Abdulkadir, A; Lienkamp, S; Brox, T; Ronneberger, O. 3D U-net: Learning dense volumetric segmentation from sparse annotation; MICCAI; 2016. 424–432.

[55]. Yu, H; Chen, X; Shi, H; Chen, T; Huang, TS; Sun, S. Motion pyramid networks for accurate and efficient cardiac motion estimation; MICCAI; 2020.
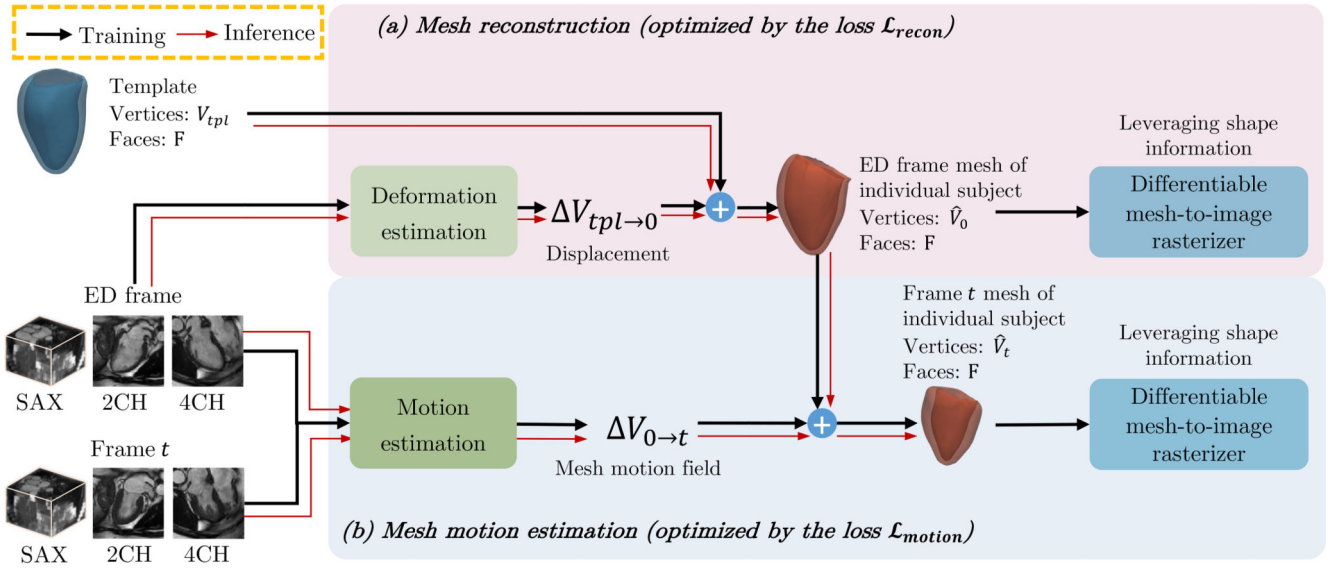
Fig. 1. An overview of the proposed method.

Panel (a) describes the mesh reconstruction module which reconstructs the ED frame mesh from a template mesh and the ED frame multi-view images. Panel (b) is the mesh motion estimation module, which takes multi-view images as input and learns 3D mesh motion field $V_{0 \to t}$. By updating the reconstructed ED frame mesh with $V_{0 \to t}$, the mesh of the t-th frame is predicted. During training, a differential mesh-to-image rasterizer is introduced to extract different 2D anatomical view planes from the predicted 3D meshes, which generates 2D soft segmentations. By comparing the predicted soft segmentations with ground truth segmentations, the rasterizer enables leveraging 2D shape information for 3D mesh reconstruction and motion estimation. Losses of each module are shown in Fig. 2 and Fig. 4, accordingly.
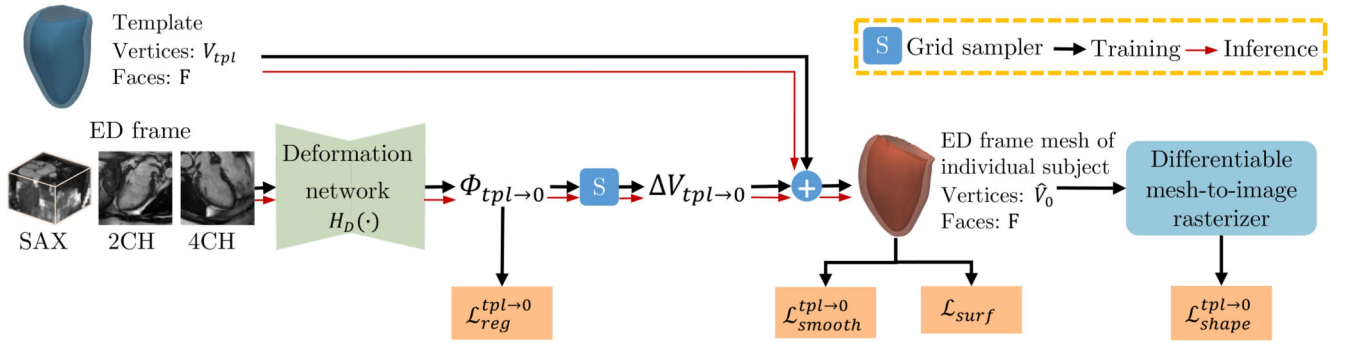
Fig. 2.

An overview of the mesh reconstruction module. This module reconstructs the ED frame mesh of individual subjects from a template mesh and multi-view images. In this module, the deformation network ($H_D(\cdot)$) predicts an intermediate voxel-wise displacement $\Phi_{tpl \to 0}$, and then $V_{tpl \to 0}$ containing the per-vertex displacement is generated by sampling from $\Phi_{tpl \to 0}$.

(a) Deformation network $H_D(\cdot)$
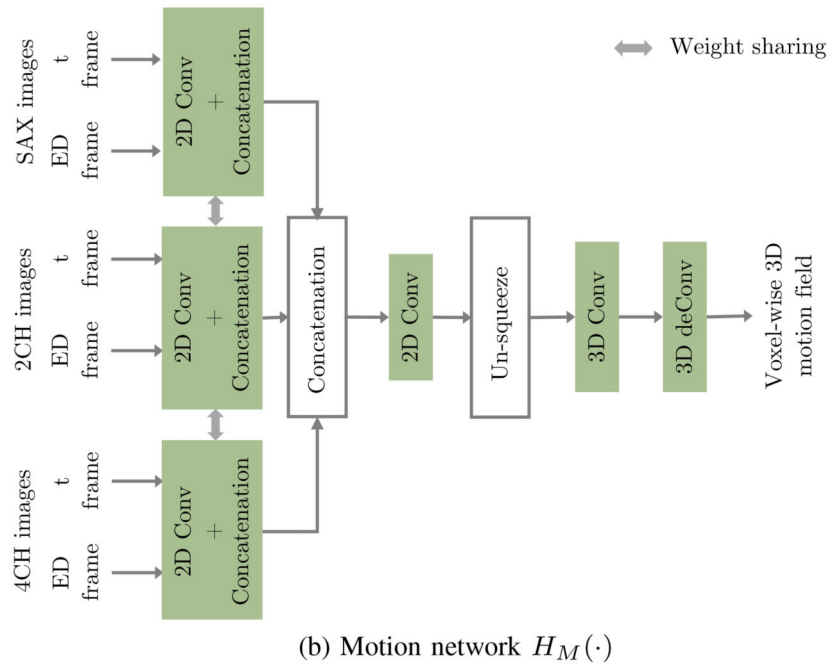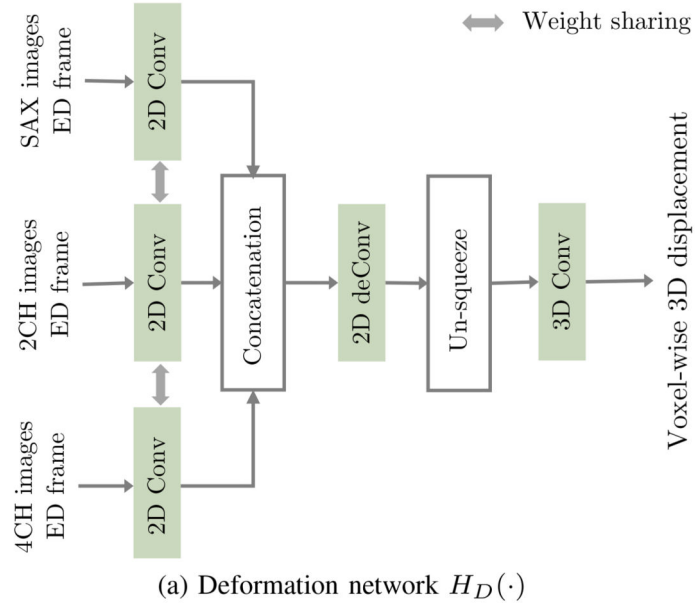
(b) Motion network $H_M(\cdot)$

Fig. 3.
A diagram of the network architecture of (a) the deformation network $H_M(\cdot)$ and (b) the motion network $H_M(\cdot)$. Here, *Conv* represents convolutional layer with Relu and batch normalization while *deConv* represents transposed convolutional layer with Relu and batch normalization. The detailed network architecture and code can be found in https:// github.com/ImperialCollegeLondon/DeepMesh.
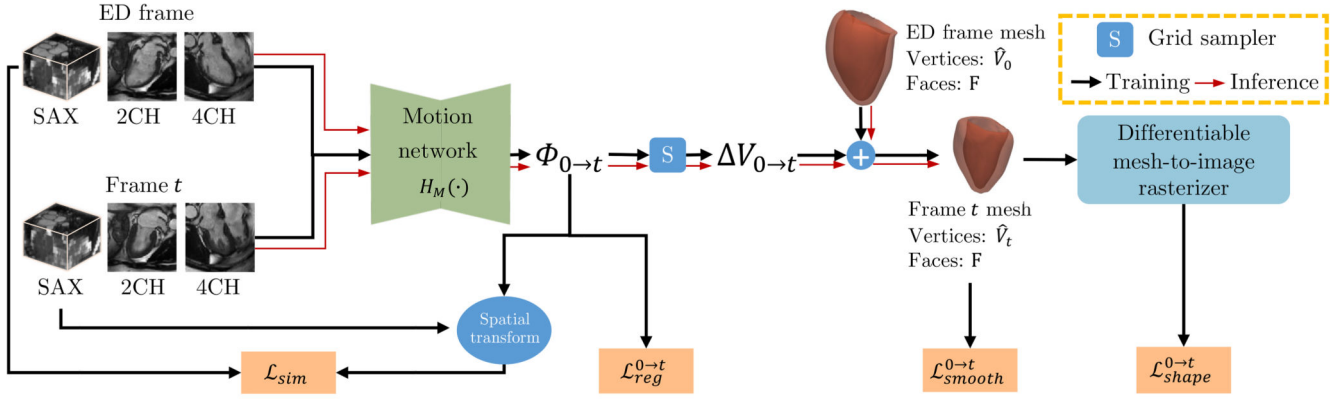
Fig. 4.

An overview of the mesh motion estimation module. This module estimates the motion of heart mesh from the ED frame to the t-th frame. It takes multi-view images of the ED frame and the t-th frame as input and learns vertex-wise 3D mesh motion field $V_{0\to t}$ via predicting an intermediate voxel-wise motion field $\Phi_{0\to t}$. By updating the myocardial mesh of the ED frame with $V_{0\to t}$, the mesh of the t-th frame is predicted.
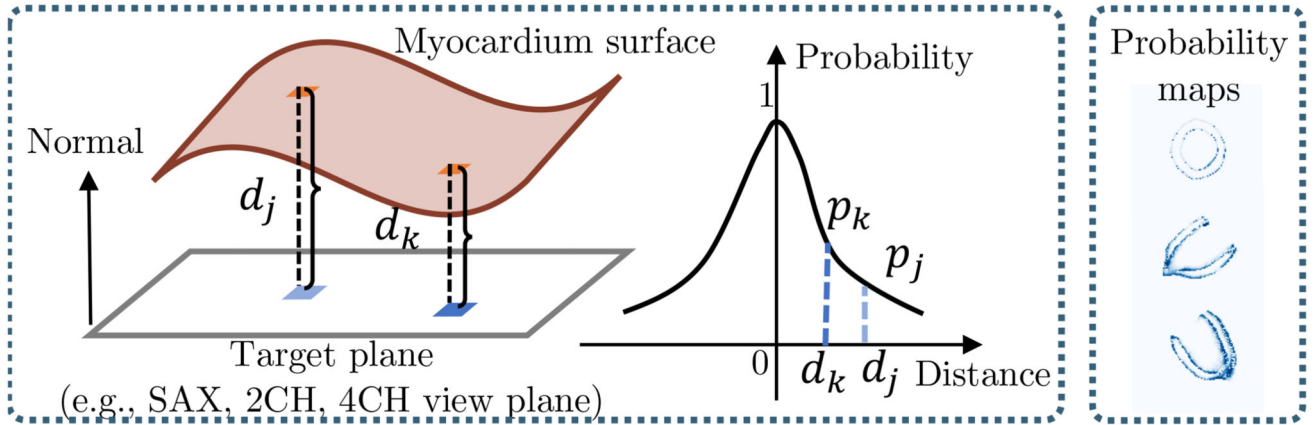
Fig. 5.
Illustration of the differentiable mesh-to-image rasterizer. This rasterizer extracts an anatomical view plane from a 3D mesh, and thus generate a 2D soft segmentation. In the left panel, d_j and d_k show the distance between a vertex of the heart surface and the target anatomical view plane. p_j and p_k refer to the probability of the vertex on the plane. The higher the distance, the lower the possibility the vertex on the plane. The right panel show examples of the obtained probability maps (2D soft contours).
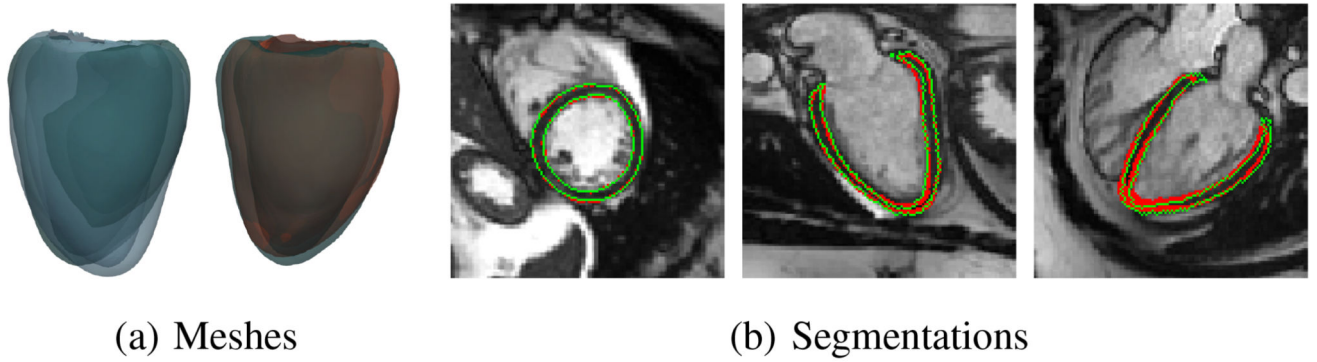
(a) Meshes   (b) Segmentations

Fig. 6. An example of ED frame mesh reconstruction.
(a) Left: ground truth mesh (Green) of a subject heart vs. the template (Blue); right: ground truth mesh (Green) vs. the reconstructed mesh (Red). (b) 2D contours on SAX, 2CH and 4CH view planes generated by rasterizing the reconstructed mesh on corresponding view planes. Red contours denote predicted results, while green contours denote ground truth.
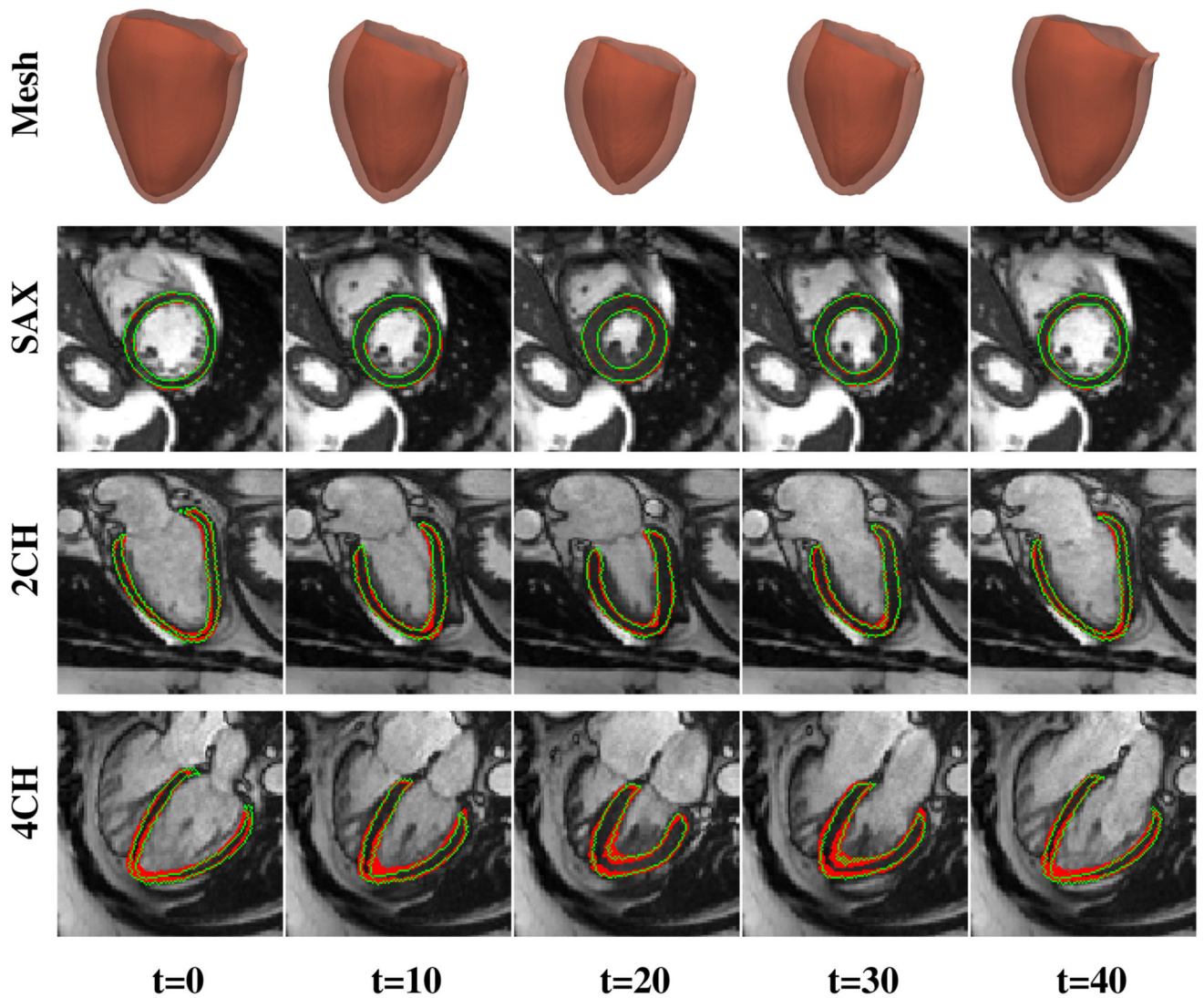
Fig. 7.

Examples of motion tracking results. The reconstructed ED frame mesh is deformed to the *t*-th frame using the estimated 3D mesh motion fields. 2D myocardium contours on SAX, 2CH and 4CH view planes (Row 2-4) are generated by extracting the corresponding planes from the predicted *t*-th frame mesh. Red contours are predicted results while green contours are ground truth.
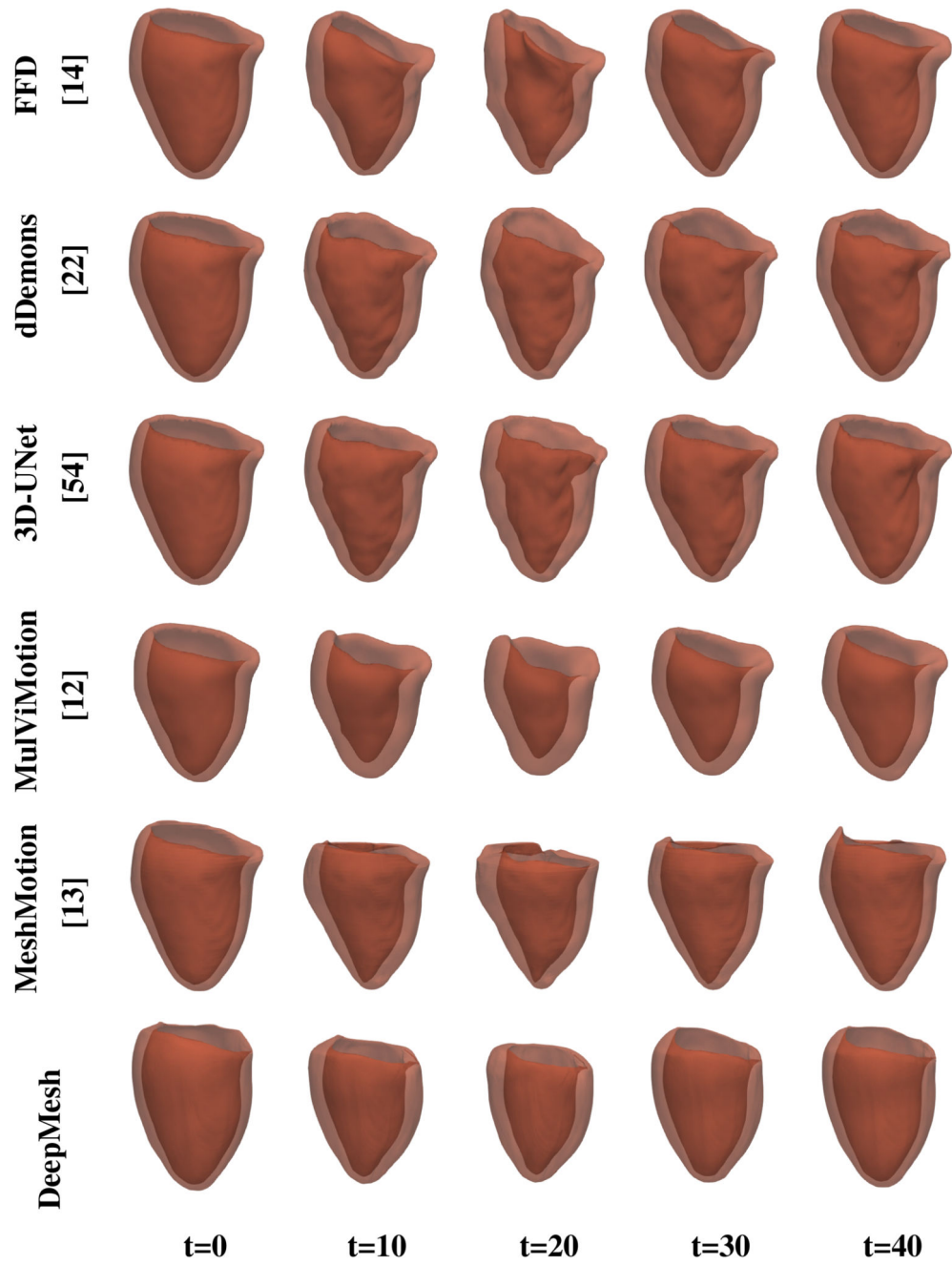
Fig. 8. Motion tracking results across the cardiac cycle using the baseline methods and the proposed method.

FFD [14]   **dDemons [22]**   **3D-UNet [54]**   **MulViMotion [12]**   **MeshMotion [13]**   **DeepMesh**   **ES frame (GT)**
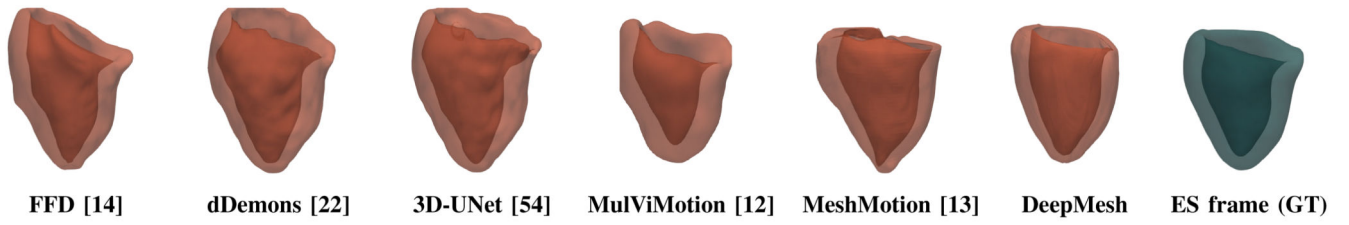
Fig. 9. Motion estimation using baseline methods and the proposed method. Green mesh is ground truth (GT) mesh of the ES frames. Red meshes are the predicted ES frame meshes based on different methods.
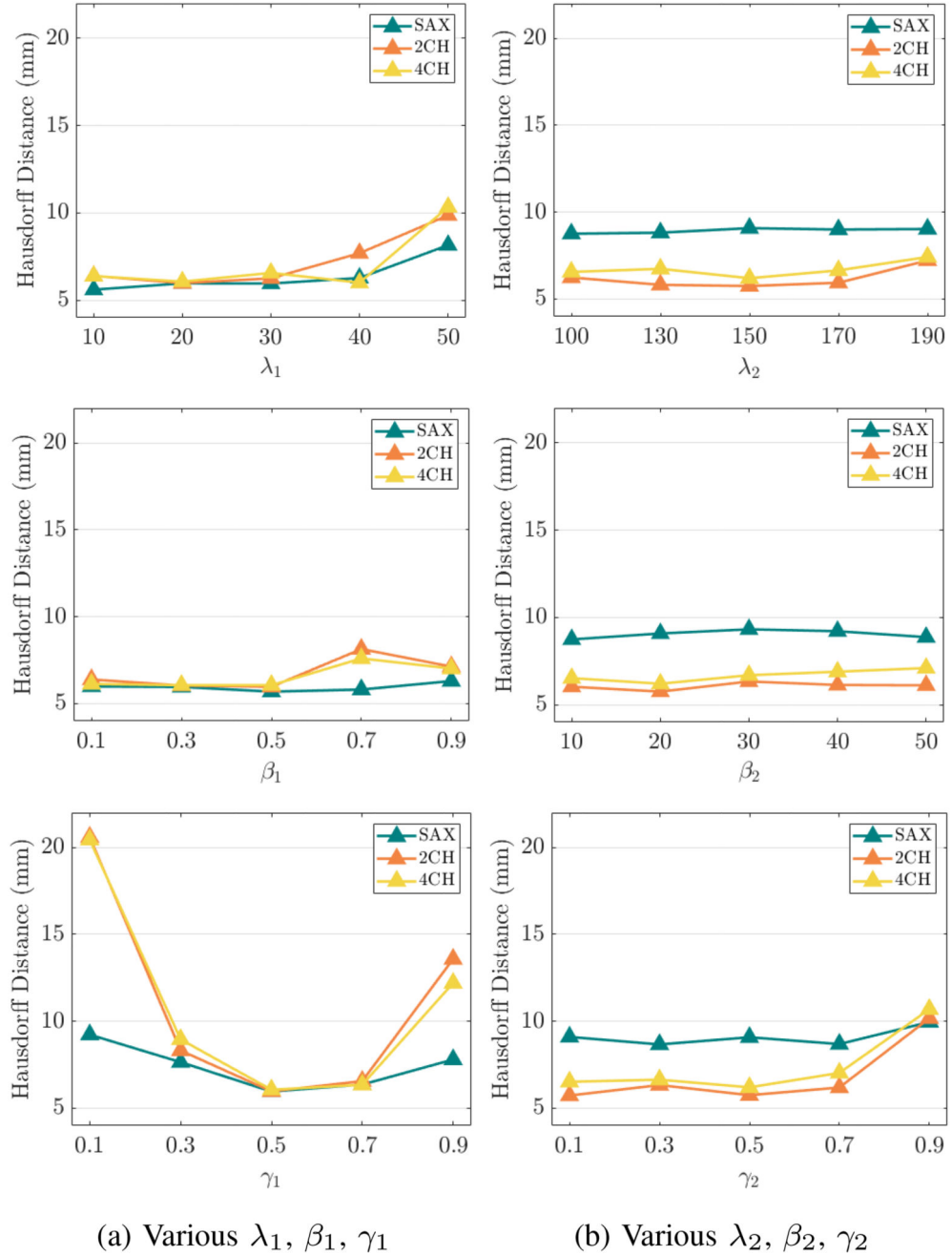
Fig. 10. Effects of varying hyper-parameters on Hausdorff distance.
(a) shows the results of using various λ_1, β_1, γ_1 for the mesh reconstruction. The final selection is $\lambda_1 = 20$, $\beta_1 = 0.5$, $\gamma_1 = 0.5$. (b) shows the results of using various λ_2, β_2, γ_2 for the mesh motion tracking. The final selection is $\lambda_2 = 150$, $\beta_2 = 20$, $\gamma_2 = 0.5$. Note that when one hyper-parameter changes, other hyper-parameters are fixed to the final selection value.
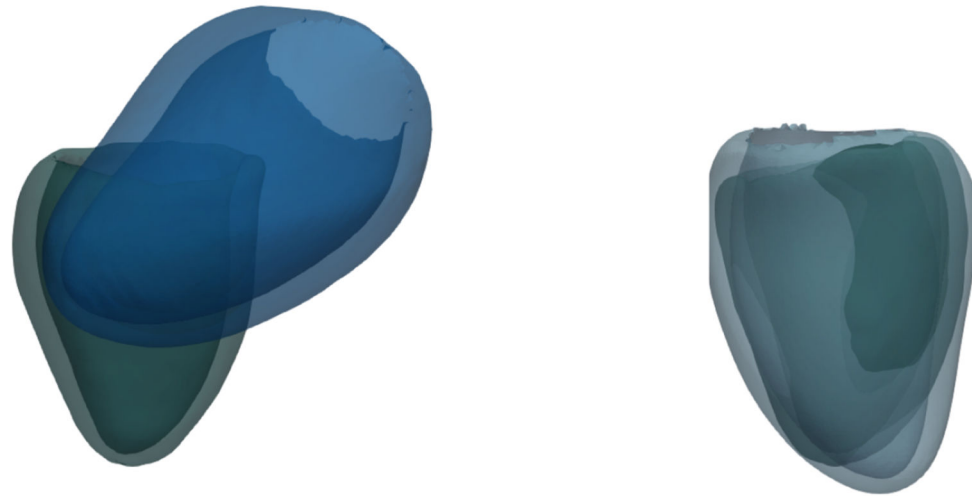
(a) Before pre-processing　　(b) After pre-processing

Fig. 11. Comparison of the template and a subject ED frame mesh.
(a) shows that the template is not in the same space as the subject mesh. (b) demonstrates that we can move the template to the subject space after data pre-processing. Green meshes are the ground truth subject mesh. Blue meshes are the template before and after data pre-processing.
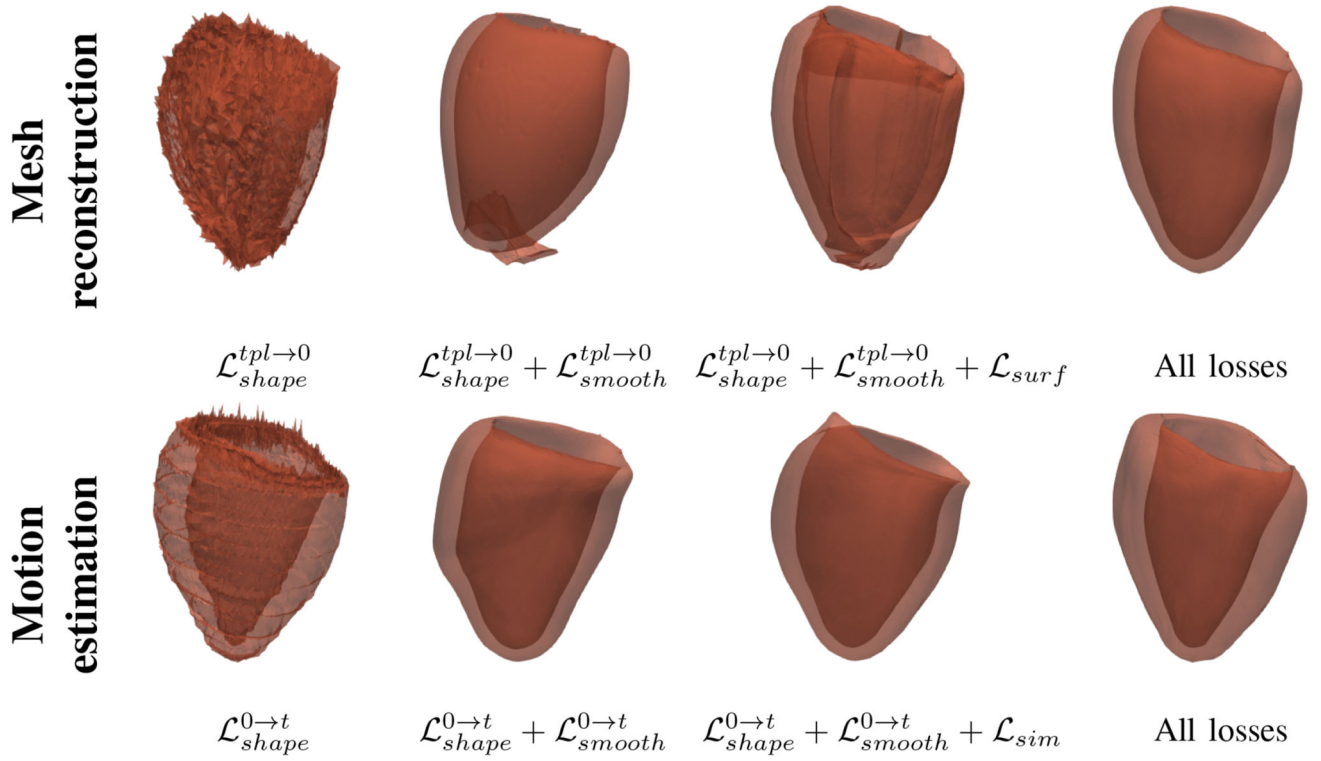
Fig. 12. Qualitative results of mesh reconstruction and mesh motion estimation with different combinations of losses. The top row shows the reconstructed ED frame mesh. The bottom row shows the estimated ES frame mesh.

Table I

Mesh reconstruction performance by comparing the predicted and ground truth 2D myocardium contours on different view planes. The results are reported as "mean (standard deviation)".

	SAX	2CH	4CH
HD (mm)	5.96 (2.14)	5.97 (1.83)	6.06 (1.94)
BoundF (%)	84.61 (5.97)	90.32 (4.94)	90.11 (3.97)

Table II

Comparison of other cardiac motion tracking methods. The results are reported as "mean (standard deviation)". ↑ indicates the higher value the better while ↓ indicates the lower value the better. Best results in bold.

Methods	Anatomical views	Mean Surface distance ↓	HD (mm) ↓			BoundF (%) ↑		
			SAX	2CH	4CH	SAX	2CH	4CH
FFD [14]	SAX	3.02(0.86)	10.31 (3.55)	15.17(4.52)	15.95(4.84)	62.15(7.48)	77.60(6.97)	77.79(7.13)
dDemons [22]	SAX	3.20(0.90)	9.71 (4.07)	15.01(3.48)	15.72(3.41)	63.67(6.92)	77.38(5.99)	80.29(5.83)
3D-UNet [54]	SAX	3.35(0.88)	8.88(3.88)	14.44(2.99)	14.83(3.57)	60.64(7.74)	74.63(6.01)	76.06(6.08)
MulViMotion [12]	SAX, 2CH, 4CH	2.39(0.79)	9.86(3.21)	9.66(3.09)	10.18(3.58)	63.65(8.42)	77.59(5.17)	76.86(6.15)
MeshMotion [13]	SAX, 2CH, 4CH	1.98(0.44)	9.73 (3.96)	7.44(4.04)	8.62(4.49)	71.49(8.82)	87.21(6.97)	84.24(6.84)
DeepMesh	SAX, 2CH, 4CH	**1.66(0.51)**	**9.08(3.86)**	**5.75(1.81)**	**6.21(2.56)**	**74.95(8.25)**	**89.26(6.97)**	**88.69(6.23)**

Table III

Mesh reconstruction with different anatomical views and different loss combinations.

Anatomical views			HD (mm) ↓		
SAX	2CH	4CH	SAX	2CH	4CH
✓			7.13 (2.25)	14.17(3.67)	14.36(3.54)
✓	✓		6.11 (2.02)	7.06(2.61)	7.41(2.75)
✓		✓	**5.50 (2.29)**	6.10(2.26)	6.14(2.32)
✓	✓	✓	5.96 (2.14)	**5.97(1.83)**	**6.06(1.94)**
Loss combinations			HD (mm) ↓		
$\mathcal{L}_{shape}^{tpl \to 0}$	$\mathcal{L}_{smooth}^{tpl \to 0}$	$\mathcal{L}_{surf}$	$\mathcal{L}_{reg}^{tpl \to 0}$	SAX	2CH	4CH
✓				7.22 (2.15)	9.63(3.38)	9.52(3.02)
✓	✓			10.95 (2.78)	14.26(3.34)	14.19(3.41)
✓	✓	✓		6.58 (2.22)	10.62(5.18)	12.66(5.79)
✓	✓	✓	✓	**5.96 (2.14)**	**5.97(1.83)**	**6.06(1.94)**

Table IV

Mesh motion estimation with different anatomical views and different loss combinations.

Anatomical views			HD (mm) ↓		
SAX	2CH	4CH	SAX	2CH	4CH
✓			**8.49 (3.67)**	6.11(2.09)	7.31(2.39)
✓	✓		8.50 (3.54)	5.88(2.41)	6.79(3.22)
✓		✓	9.38 (4.17)	6.37(1.78)	6.27(2.29)
✓	✓	✓	9.08(3.86)	**5.75(1.81)**	**6.21(2.56)**
Loss combinations			HD (mm) ↓		
$\mathscr{L}_{shape}^{0 \to t}$	$\mathscr{L}_{smooth}^{0 \to t}$	$\mathscr{L}_{sim}$ $\mathscr{L}_{reg}^{0 \to t}$	SAX	2CH	4CH
✓			**8.97 (3.35)**	7.20(2.00)	7.34(2.13)
✓	✓		11.08 (4.07)	10.56(2.23)	10.43(2.26)
✓	✓	✓	9.92 (3.44)	8.19(2.52)	8.67(3.21)
✓	✓	✓ ✓	9.08(3.86)	**5.75(1.81)**	**6.21(2.56)**