Bidirectional Recurrence for Cardiac Motion Tracking with Gaussian Process Latent Coding

Jiewen Yang **Yiqun Lin** **Bin Pu** **Xiaomeng Li**✉
The Hong Kong University of Science and Technology
{jyangcu, ylindw}@connect.ust.hk {eebinpu, eexmli}@ust.hk

Abstract

Quantitative analysis of cardiac motion is crucial for assessing cardiac function. This analysis typically uses imaging modalities such as MRI and Echocardiograms that capture detailed image sequences throughout the heartbeat cycle. Previous methods predominantly focused on the analysis of image pairs lacking consideration of the motion dynamics and spatial variability. Consequently, these methods often overlook the long-term relationships and regional motion characteristic of cardiac. To overcome these limitations, we introduce the **GPTrack**, a novel unsupervised framework crafted to fully explore the temporal and spatial dynamics of cardiac motion. The GPTrack enhances motion tracking by employing the sequential Gaussian Process in the latent space and encoding statistics by spatial information at each time stamp, which robustly promotes temporal consistency and spatial variability of cardiac dynamics. Also, we innovatively aggregate sequential information in a bidirectional recursive manner, mimicking the behavior of diffeomorphic registration to better capture consistent long-term relationships of motions across cardiac regions such as the ventricles and atria. Our GPTrack significantly improves the precision of motion tracking in both 3D and 4D medical images while maintaining computational efficiency. The code is available at: https://github.com/xmed-lab/GPTrack.

1 Introduction

Cardiac motion tracking from Cardiac Magnetic Resonance Imaging (MRI) and Echocardiograms is crucial in quantitative cardiac image processing. These imaging techniques provide comprehensive image sequences that cover an entire heartbeat cycle, allowing for detailed analysis of cardiac dynamics. Conventional non-parametric cardiac motion tracking approaches, such as B-splines [1], Demons algorithms [2] and optical-flow based methods [3, 4, 5], are commonly utilized due to their flexibility and ability to align detailed structures within images. However, these methods face significant challenges in motion tracking because they lack topology-preserving constraints and temporal coherence. The diffeomorphic registration method [6, 7, 8], which formulates the registration process as a group of diffeomorphisms in Lagrangian dynamics, is a nice candidate for topology-preserving motion tracking. However, traditional optimization-based diffeomorphic registration methods are computationally intensive and sensitive to noises, hindering their applications in efficient cardiac motion tracking.

Current deep learning-based techniques [9, 10, 11, 12, 13, 14, 15, 16] employ these advanced imaging modalities to register image pairs within the same patient, and some [14, 15, 16] adopt the diffeomorphic routine and learn the Lagrangian strain to describe the motion relationship between the reference frame and subsequent frames, aggregating dynamic information into consecutive cardiac motions as Lagrangian displacements. Although the above diffeomorphic methods better model the dynamic and continuous nature of cardiac motion, they still have room for improvement in handling the long-term temporal relationship in videos. For example, the approach [14] requires segmentation

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-1098

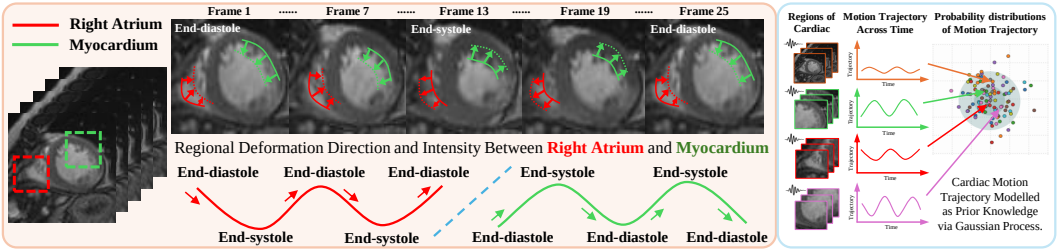

Figure 1: **Regional Motions in Cardiac:** The left sequential MRI frames within a heartbeat cycle illustrate that motion direction and intensity are completely different between the right atrium and myocardium during End-diastole and End-systole. **Formulate Cardiac Motion as Prior Knowledge:** The right figure depicts the regions of motion trajectory across the heartbeat cycle, alongside the probability distributions of motion trajectory. Curves (Middle) are the motion trajectory changes of different MRI sequences (Left). Highlighting the cardiac motion trajectory that follows a certain pattern can be modelled as prior knowledge via the Gaussian Process (Right).

annotation to generate dense motion trajectories and calculate Lagrangian strain. Additionally, the methods outlined in [15, 16] are prone to error accumulation as Lagrangian displacements are integrated without regularizing temporal variations. In spatial views, while the global trajectory flow may follow a specific pattern, significant variations exist within each region regarding the phases, amplitudes, and intensities of the motion. For example, Figure 1 shows regions of the Right Atrium (red) and Myocardium (green) performing the opposite trajectories during the heartbeat cycle. Conversely, similar regions across different cases exhibit consistent motions. Hence, ignoring the regional scale may lead to a fragmented understanding of cardiac motion, underscoring the need for more nuanced analytical approaches. Furthermore, as shown in the right of Figure 1, the deformation is bounded in the space of periodically specific human cardiac motion variation.

To leverage discussed temporal and spatial information for cardiac motion tracking, we propose a novel framework named **GPTrack**. Our GPTrack has several appealing facets: **1)** GPTrack employs the Gaussian Process (GP) to formulate the consistent temporal patterns in the latent space of diffeomorphic frameworks, promoting the consistency of cardiac motion; **2)** GPTrack utilizes position information in the latent space to encode the statistics of the sequential Gaussian process, by which we model the region-specific motion and obtain a more precise estimation related to cardiac motion; **3)** GPTrack leverages the inherent temporal continuity in cardiac motion by aggregating long-term relationships through forward and backward video flows, which mimics the forward-backward manner of classical diffeomorphic registration framework [6]. To evaluate the performance of GPTrack in cardiac motion tracking, we conduct experiments based on 3D Echocardiogram videos [17, 18] and 4D temporal MRI image [19]. Results in Tables 1, 2 and 3, show the GPTrack enhance the accuracy of motion tracking performance in a clear margin, without substantially increasing the computational cost in comparison to other state-of-the-art methods. **Our contributions are summarized as follows:**

1. We propose a novel cardiac motion tracking framework named the **GPTrack**. This framework employs the *Gaussian Process* (GP) to promote temporal consistency and regional variability in compact latent space, establishing a robust regularizer to enhance cardiac motion tracking accuracy.

2. The GPTrack framework is designed to capture the long-term relationship of cardiac motion via a bidirectional recursive manner, and its forward-backward manner mimics the workflows of the classical diffeomorphic registration framework. By this approach, our method provides a more accurate and reliable estimation of cardiac motion.

3. Our GPTrack framework achieves state-of-the-art performance on both 3D Echocardiogram videos and 4D temporal MRI datasets, maintaining comparable computational efficiency. The results demonstrate that our method adapts effectively across different medical imaging modalities, proving its utility in different clinical settings.

2 Related Work

2.1 Cardiac Motion Tracking via Non-parametric Registration Approach

Extensive works have been proposed to address registration by optimising within the space of displacement vector fields. Models related to elastic matching were proposed by [20, 21]. [22] utilized

statistical parametric mapping for improvement. Techniques incorporating free-form deformations with B-splines and Maxwell demons were adapted by [1] and [2], respectively. The Harmonic phase-based method, which utilizes spectral peaks in the Fourier domain for cardiac motion tracking, was introduced by [23]. This method calculates phase images from the inverse Fourier transforms and is specifically exploited in the analysis of tagged MRI. Popular formulations [6, 8, 24, 7] introduce topology-preserved diffeomorphic transforms. In the realm of diffeomorphic registration, inverse consistent diffeomorphic deformations have been estimated by [6] and [7]. Syn [24] proposed standard symmetric normalization. RDMM [8] considered regional parameterization based on Large Deformation Diffeomorphic Metric Mapping [6]. Despite their remarkable success in computational anatomy studies, these approaches are also time-consuming and susceptible to noise.

2.2 Cardiac Motion Tracking with Deep-Learning based Registration Method

Recent advancements in medical image registration have increasingly leveraged Deep Learning technologies. Pioneering studies [14, 25, 26, 27, 28] utilize ground truth displacement fields obtained by simulating deformations and deformed images, typically estimated using non-parametric methods. These approaches, however, may be limited by the types of deformations they can effectively model, which can affect both the quality and accuracy of the registration. Unsupervised methods, as discussed by [9, 10, 15, 29, 30, 31, 32] have shown promise by learning deformation through the warping of a fixed image to a moving image using spatial transformation functions [33]. These methods have been extended to include deformable models for single directional deformation field tracking [9, 26, 34] and diffeomorphic models for stationary velocity fields [32, 35, 36]. Further application of diffeomorphic models to cardiac motion tracking has been explored by [13, 15, 16, 30, 31]. These models predict motion fields that are both differentiable and invertible, ensuring one-to-one mappings and topology preservation. Recent studies in denoising diffusion probabilistic models (DDPM), such as [11] and [12], have achieved considerable success in registration tasks. However, DDPM-based methods face challenges in building temporal connections and demand substantial computational resources. The DL-based optical flow (OF) methods [37, 38, 39, 40] apply widely in nature image motion tracking. However, as illustrated in [7, 15, 16], due to annotations requirements and photometric constraints, they cannot be adopted in unsupervised cardiac motion tracking in medical image domains *(See Section A1 in Appendix for detailed discussion)*.

3 Methodology

3.1 Diffeomorphic Tracking of Cardiac Motion

Diffeomorphic motion tracking and registration techniques are widely used in medical image analysis because they seek topology-preserving mapping between source and target images [6, 8]. Formally, given the source image x_0 and target image x_1, the diffeomorphic registration aims at a family of differentiable and invertible mappings $\{\phi_t\}_{t\in[0,1]}$ with the boundary condition $\phi_0 = \mathrm{Id}$ and $\phi_1(x_0) = x_1$, where Id is the identity mapping. The diffeomorphism ϕ_t can be parameterized as its derivatives (velocity field) $\mathbf{v}_t$ as follows:

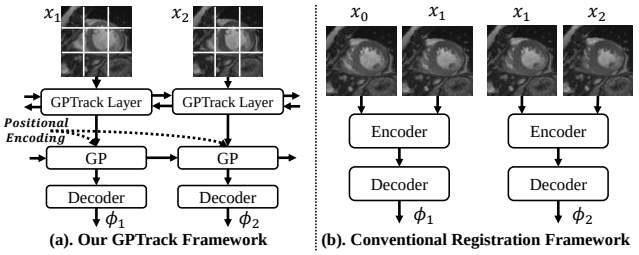

Figure 2: Comparsion between our GPTrack (a) and conventional registration framework (b).

$$\frac{d\phi_t}{dt} = \mathbf{v}_t(\phi_t) := \mathbf{v}_t \circ \phi_t \Longleftrightarrow \phi_t = \phi_0 + \int_0^t \mathbf{v}_s(\phi_s)ds, \quad s \in [0,1], \tag{1}$$

where $\circ$ is the composition operator. For numerical implementation, the associative property of the diffeomorphism group indicates $\phi_{t_1+t_2} = \phi_{t_1} \circ \phi_{t_2}$, and the integral of Equation 1 can be approximated by $\phi_{t+\delta} \approx \phi_t + \mathbf{v}_t\delta$ for $t \in [0,1)$ and small enough δ. In this study, we follow the parameter settings of [7, 15, 16, 35, 36] and take $\delta = \frac{1}{2^N}$, $N = 7$ to discretize the path of diffeomorphic deformation.

https://doi.org/10.52202/079017-1098

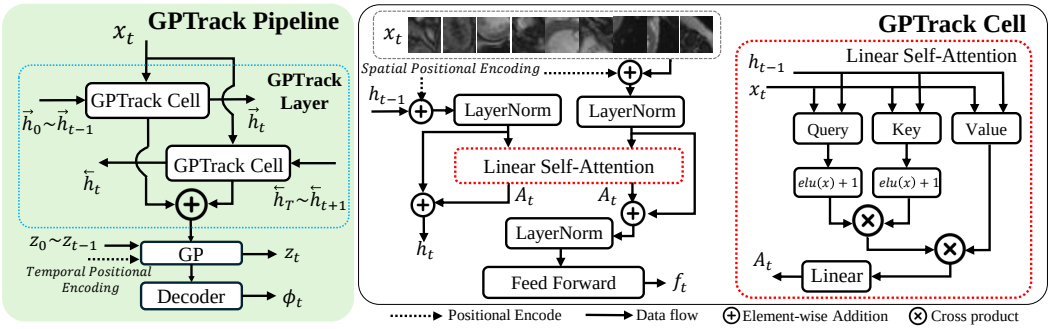

Figure 3: The overview pipeline of GPTrack (one layer). The x, h, $\dot{h}$ and z denote input, forward hidden states, backward hidden states and latent coordinates. Feature $\vec{f}_t$ with probabilistic prior on the latent space via Gaussian Process then enters the decoder to predict the motion field ϕ. Subscript t denotes the t-th position in total T moments. $elu(\cdot)$ represent the exponential linear units [42].

3.2 Motion Tracking with Gaussian Process Latent Coding

Our proposed method adopts the generative variational autoencoder (VAE) framework as the backbone of a diffeomorphic tracking network, as suggested by previous methods [15, 16, 36]. However, as shown in Figure 2, the first difference between our method and other methods is that ours allows the registration network to aggregate the spatial information temporally, both forward and backward (see Figure 2(a)). The conventional approaches [9, 10, 15, 16, 24, 31] only conduct between two adjacent frames, which ignore the relationship of long-term dependency of the cardiac motion (see Figure 2(b)). Secondly, despite the large deformation through the path of diffeomorphism, the space of periodically specific human cardiac motion variation is bounded. Though methods [14, 15, 16] use Lagrangian strain to formulate the continuous dynamic of cardiac motion, however, without considering the motion consistency between two adjacent state spaces, the Lagrangian strain is prone to error accumulation and degrade the tracking performance. To address this problem, we take the simple yet efficient Bayesian approach, which employs the Gaussian Process (GP) to model cardiac motion dynamics in the compact feature space, predicting more consistent motion fields over dynamics parameters. Our proposed GPTrack can also be easily extended to other modalities or motion-tracking tasks, such as 3D Echocardiogram videos and 4D cardiac MRI.

In this paper, we follow the research [41] that employs the recursive manner in transformer for sequential data. As shown in Figure 3 left, the GPTrack pipeline comprised the GPTrack layer for feature extraction, the Gaussian Process (GP) layer for modelling the cardiac motion dynamics, and the Decoder for motion field estimation. Given the sequential 4D inputs $\{x_t\}_{t=1}^{T}, x_t \in \mathbb{R}^{H\times W\times D\times 1}$, where H, W, D, T denote the height, width, depth, length of the input. For each x_t, we first decompose it to P non-overlapping patches of shape $p \times p \times p$, where $P = \frac{H}{p} \times \frac{W}{p} \times \frac{D}{p}$ and $p, \frac{H}{p}, \frac{W}{p}, \frac{D}{p} \in \mathbb{Z}^{+}$. We then embed each patch as a feature with C channels via embedding layers and disentangle patches with the dimension of $\mathbb{R}^{P\times C}$ from x_t. The GPTrack layer then takes the x, both forward and backward hidden states $\vec{h}, \overleftarrow{h} \in \mathbb{R}^{P\times C}$ as the input, then predicts the motion field ϕ via the decoder after the GP layer. Note that the initial hidden states of forward $\vec{h}_0$ and backward $\overleftarrow{h}_0$ are set as zero.

3.3 Bidirectional Forward-Backward Recursive Cell

The GPTrack layer consists of two independent GPTrack cells that respond to forward and backward computation. Similar to the [41, 43], adapting the hidden state to maintain and aggregate the sequential information allows the input with variable length. Meanwhile, the conventional convolutional neural network or vision in the transformer-based method is limited by the fixed input length. Furthermore, medical images such as Echocardiogram videos usually consist of hundreds of frames that cover multiple heartbeat cycles. Hence, parallel computing all frames requires a large amount of computational consumption, which hinders the application in real scenarios limited by low-computational devices. To this end, as shown in Tables 1 and 2, our GPTrack is able to formulate the variable temporal information while maintaining the comparable computational cost.

Using the forward GPTrack cell shown in the right of Figuer 3 as an example, the x_t and $\vec{h}_{t-1}$ followed by the addition of learnable position encoding $\text{pos}_t \in \mathbb{R}^{P\times C}$ are respectively normalized by Layer Normalization. The linear self-attention then computes the attentive weight $A_t \in \mathbb{R}^{P\times C}$ of combined x_t and $\vec{h}_{t-1}$. The above operations can be formulated as follows:

$$A_t = (\delta(\mathcal{W}_Q x)+1)(\delta(\mathcal{W}_K x)+1)^{\mathsf{T}}\mathcal{W}_V x;\ x = \text{LN}(x_t+\text{pos}_t)\oplus\text{LN}(\vec{h}_{t-1}+\text{pos}_t) \in \mathbb{R}^{P\times 2C}, \quad (2)$$

where $\mathcal{W}_Q, \mathcal{W}_K, \mathcal{W}_V \in \mathbb{R}^{2C\times C}$ are learnable weights of different projection layers named query, key and value, $\delta(\cdot)$ represent the exponential linear units $elu(\cdot)$ [42], $\text{LN}(\cdot)$ and $\oplus$ are the layer normalization and the concatenation operation, respectively.

In order to raise the descendant hidden state $\vec{h}_t$ that aggregates information before $t+1$ moment. The attentive weight A_t then conducts the element-wise addition with ancestral positional encoded $\vec{h}_{t-1}$. Additionally, the A_t takes the positional encoded x_t as the residual connection and applies the addition operation. The Feed Forward Network denoted as $\text{FFN}(\cdot)$, is then introduced to output the feature of t-th moment. The formulation can be written as follows:

$$\vec{h}_t = A_t + \text{LN}(\vec{h}_{t-1}+\text{pos}_s) \in \mathbb{R}^{P\times C},\ \vec{f}_t = \text{FFN}(\text{LN}(A_t+\text{LN}(x_t+\text{pos}_s))) \in \mathbb{R}^{P\times C}. \quad (3)$$

In the Bidirectional Forward-Backward Recursive Cell, both forward and backward share the same computation processes through the GPTrack cell. The only difference between the two directions is that the forward process starts from the first moment x_0 of input while the backward starts from the last moment x_T. Hence, the feature f_t in t-th moment is formulated as $f_t = \vec{f}_t + \overleftarrow{f}_t,\ f_t \in \mathbb{R}^{P\times C}$, where $\vec{f}_t$ aggregate the forward information from 0 to t, while the $\overleftarrow{f}_t$ aggregate the backward information from the last frame to the t-th frame.

3.4 Gaussian Process in Cardiac Motion Tracking

The primary objective of integrating the Gaussian Process (GP) is to establish a probabilistic prior in the latent space that incorporates prior knowledge. Specifically, it posits that cardiac motion across different individuals within the same region should yield similar motion fields in latent space encodings. Furthermore, as illustrated in Figure 1, cardiac structures within an individual that are spatially distant or exhibit motions in opposite directions should consistently adhere to the periodic pattern of motion.

Initially, we define a covariance (kernel) function for the GP layer as depicted in Figure 3. We design the prior for the latent space processes to be stationary, mean square continuous, and differentiable in sequential motion fields. This design stems from our expectation that the latent functions should model cardiac motion more prominently than visual features. Consequently, we anticipate the latent space to manifest continuous and relatively smooth behaviour. To this end, we employ the isotropic and stationary Matern kernel (refer to Equation 4) to fulfil the required covariance function structure:

$$\kappa(x_t, x_{t-1}) = \sigma\frac{2^{1-\nu}}{\Gamma(\nu)}(\sqrt{2\nu}\frac{D(x_t,x_{t-1})}{l})^{\nu}K_{\nu}(\sqrt{2\nu}\frac{D(x_t,x_{t-1})}{l}), \quad (4)$$

where $\nu, \sigma, l > 0$ are the smoothness, magnitude and length scale parameters, K_ν is the modified Bessel function, and $D(\cdot,\cdot)$ denotes the distance metric between features of two consecutive motion fields. Our goal is to formulate cardiac motion as robust prior knowledge applicable to unseen data. To address this, we propose a position-related distance measurement.

As outlined in section 3.3 and referenced in [44], we utilize a learnable parameter $\dot{\text{pos}} \in \mathbb{R}^{P\times C}$ as the spatial positional encoding for each region, which provides relative or absolute positional information about the decomposed patches. To capture the periodic temporal positional information of cardiac motion, distinct from the spatial encoding $\dot{\text{pos}}$, we apply sine and cosine functions of various frequencies as temporal encoding $\{\tilde{\text{pos}}_t\}_{t=1}^{T}, \tilde{\text{pos}}_t \in \mathbb{R}^{P\times C}$ for each moment. The overall positional encoding at moment t is formulated as:

$$\text{pos}_t = \dot{\text{pos}}\cdot\tilde{\text{pos}}_t \in \mathbb{R}^{P\times C},\ \text{where } \tilde{\text{pos}}_t(t,i) = \mathbb{I}_{(i=2k)}\sin(t\cdot n^{-\frac{2k}{C}}) + \mathbb{I}_{(i=2k+1)}\cos(t\cdot n^{-\frac{2k}{C}}). \quad (5)$$

The i denotes the i-th position of C channels, and n is the scaling factor. We assign independent GP priors to all values in $\{z_t\}_{t=1}^{T}, z_t \in \mathbb{R}^{P\times C}$ to disseminate temporal information between frames

https://doi.org/10.52202/079017-1098

in the sequence. During this process, we regard the sequential output $\{f_t\}_{t=1}^{T}$ of GPTrack as noise-corrupted versions of the ideal latent space encodings, formulating the inference as the following GP regression model with noise observations:

$$z_t \sim \mathrm{GP}\left(\mu(\mathrm{pos}_t), \kappa\left(\mathrm{pos}_{t-1}, \mathrm{pos}_t\right)\right),\ f_t = z_t + \epsilon_t,\ \epsilon_t \sim \mathcal{N}(0, \sigma^2),$$

where σ^2 is the noise variance of the likelihood model set as the learnable parameter in GPTrack. The above Gaussian process can be treated as the temporal sequence with intrinsic Markov property, and we adopt the methodology of connecting the Gaussian process with state space model [45] to decrease its computational complexity from $O(T^3)$ to $O(T)$, where T is the number of time step. Concretely, the Gaussian process of Equation 6 corresponds to the following linear stochastic differential equation:

$$\frac{d}{dt}\mathbf{z}(t) = \mathbf{A}\mathbf{z}(t) + \mathbf{b}w(t),\ \ f(t) = \mathbf{h}^{\mathsf{T}}\,\mathbf{z}(t) + \epsilon(t),\ \epsilon(t) \sim \mathcal{N}(0, \sigma^2), \tag{6}$$

with the solution as:

$$\begin{aligned} \mathbf{z}(t) &= \exp^{(t-r)\mathbf{A}}\,\mathbf{z}(r) + \int_r^t \exp^{(t-s)\mathbf{A}}\,\mathbf{b}w(s)ds,\ \forall r < t, \\ f(t) &= \mathbf{h}^{\mathsf{T}}\mathbf{z}(t) + \epsilon(t),\ \epsilon(t) \sim \mathcal{N}(0, \sigma^2), \end{aligned} \tag{7}$$

where $\mathbf{z}(t) := (z(t), \frac{d}{dt}z(t))$, $w(t)$ is the zero-mean Gaussian random process, and $\mathbf{h} := (0, 1)^{\mathsf{T}}$ is used for modelling observation model. The state transition matrix (vector) $\mathbf{A}$ and $\mathbf{b}$ can be calculated from the covariance function κ of the Gaussian process. For the Matern kernel shown in Equation 4, we take $\nu = \frac{3}{2}$, and corresponding state transition matrix $\mathbf{A}$ and vector $\mathbf{b}$ [46] read as:

$$\mathbf{A} = \begin{pmatrix} 0, & 1 \\ -\frac{3}{l^2}, & -\frac{2\sqrt{3}}{l} \end{pmatrix},\ \mathbf{b} = (0, 1)^{\mathsf{T}}. \tag{8}$$

Then we can discretize Equation 7 and get its weakly equivalent state-space model of Equation as:

$$\mathbf{z}_t = \mathbf{\Phi}_t\,\mathbf{z}_{t-1} + \mathbf{n}_t,\ f_t = \mathbf{h}^{\mathsf{T}}\mathbf{z}_t + \epsilon_t,\ \epsilon(t) \sim \mathcal{N}(0, \sigma^2),\ t = 1, \ldots, T, \tag{9}$$

where $\mathbf{\Phi}_t = \exp^{D(\mathrm{pos}_t, \mathrm{pos}_{t-1})\mathbf{A}}$, $\mathbf{n}_t \sim \mathcal{N}(0, \mathbf{\Phi}_t\mathbf{b}Q_w(t-1, t)\mathbf{b}^{\mathsf{T}}\mathbf{\Phi}_t^{\mathsf{T}})$, and $Q_w(t, t-1)$ is the covariance of $w(t)$. Given the initial value $\mathbf{z}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \mathbf{\Sigma}_0)$ with $\boldsymbol{\mu}_0 = 0$ and $\mathbf{\Sigma}_0 = \mathrm{diag}(\frac{\sigma^2}{2}, \frac{3\sigma^2}{l^2})$, we can sequentially calculate the posterior distribution $\mathbf{z}_t|f_{1:t-1} \sim \mathcal{N}(\overline{\boldsymbol{\mu}}_t, \overline{\mathbf{\Sigma}}_t)$ and $\boldsymbol{z}_t|f_{1:t} \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{\Sigma}_t)$ using update criterion of Kalman filter for state space model [47] as:

$$\begin{aligned} &\overline{\boldsymbol{\mu}}_t \leftarrow \mathbf{\Phi}_t\overline{\boldsymbol{\mu}}_{t-1}, && \overline{\mathbf{\Sigma}}_t \leftarrow \mathbf{\Phi}_t\overline{\mathbf{\Sigma}}_{t-1}\mathbf{\Phi}_t^{\mathsf{T}} + \mathbf{\Sigma}_0 - \mathbf{\Phi}_t\mathbf{\Sigma}_0\mathbf{\Phi}_t^{\mathsf{T}}, \\ &\boldsymbol{\mu}_t \leftarrow \overline{\boldsymbol{\mu}}_t + \mathbf{k}_t(f_t - \mathbf{h}^{\mathsf{T}}\overline{\boldsymbol{\mu}}_t), && \mathbf{\Sigma}_t \leftarrow \overline{\mathbf{\Sigma}}_t - \mathbf{k}_t\mathbf{h}^{\mathsf{T}}\overline{\boldsymbol{\mu}}_t,\ t = 1, \ldots, T, \end{aligned} \tag{10}$$

where $\mathbf{k}_t := \frac{\overline{\mathbf{\Sigma}}_t\mathbf{h}}{\mathbf{h}\overline{\mathbf{\Sigma}}_t\mathbf{h} + \sigma^2}$ is the optimal Kalman gain at time t. The output of the GP layer in t-th moment thus can be formulated as $z_t^{GP} = \mathrm{ReLU}(\mathbf{k}_t z_t)$, where ReLU is the activation function. In the final, the t-th motion field θ_t is obtained by decoder from the z_t^{GP}.

3.5 Overall Loss of Tracking a Time Sequence of Cardiac Motion

The decoder takes the Gaussian-process coding $\{z_t^{GP}\}_{t=1}^{T}$ as velocity filed to composite the diffeomorphic motion field $\boldsymbol{\phi}$ according to the criterion of Section 3.1. Here, we adopt the training loss of [15, 16], which minimizes the integration of four components summarized as follows: **a)** Dissimilarities of tracking results between adjacent states from both forward and backward; **b)** Smoothness of motion fields between adjacent states from both forward and backward; **c)** Dissimilarities of tracking results between the start state and each state; **d)** Smoothness of motion fields between the start state and each state. The overall loss function $\mathcal{L}$ is formulated as:

$$\sum_{t=1}^{T-1}[\underbrace{\mathcal{L}_{kl}(x_t, x_{t+1})}_{\mathbf{a})} + \alpha_1(\underbrace{\mathcal{L}_{sm}(\boldsymbol{\phi}_{t:t+1}) + \mathcal{L}_{sm}(\boldsymbol{\phi}_{t+1:t})}_{\mathbf{b})}) + \alpha_2\underbrace{\mathcal{L}_{nc}(x_{t+1}, x_1 \circ \boldsymbol{\phi}_{0:t+1})}_{\mathbf{c})} + \alpha_3\underbrace{\mathcal{L}_{sm}(\boldsymbol{\phi}_{1:t+1})}_{\mathbf{d})}],$$

where α_1, α_2 and α_3 are loss weights, and $\phi_{t_1:t_2}$ is the motion field from state t_1 to t_2. $\mathcal{L}_{kl}(x_t, x_{t+1}) = \mathbb{KL}(q(z_t^{GP}|x_t; x_{t+1})||p(z_t^{GP}|x_t; x_{t+1})) + \mathbb{KL}(q(z_t^{GP}|x_{t+1}; x_t)||p(z_t^{GP}|x_{t+1}; x_t))$ is the summation of forward and backward VAE losses with latent coding z_t^{GT}, posterior distribution q and conditional distribution p, L_{nc} is the negative normalized local cross-correlation metric, and $\mathcal{L}_{sm}(\boldsymbol{\phi}) = ||\nabla\boldsymbol{\phi}||_2^2$ is the ℓ_2-total variation metric.

4 Experiment

4.1 Datasets

CardiacUDA [17]. The CardiacUDA dataset collected from two medical centers consists of 314 echocardiogram videos from patients. The video is scanned from the apical four-chamber heart (A4C) view. In this paper, we conduct training and validation in the A4C view that consists of 314 videos with 5 frame annotations in the Left/Right Ventricle and Atrium (LV, LA, RV, RA). For testing, we report our results in 10 videos with full annotation provided by the CardiacUDA.

CAMUS [18]. The CAMUS dataset provides pixel-level annotations for the left ventricle, myocardium, and left atrium in the Apical two-chamber view, which consists of 500 echocardiogram videos in total. There are 450 subjects in the training set with 2 frames annotated in the Left Ventricle (LV), Left Atrium (LA) and Myocardium (Myo) in the end-diastole (ED) and end-systole (ES) of the heartbeat cycle. The remaining 50 subjects without any annotation masks belong to the testing set.

ACDC [19]. The ACDC dataset consists of 100 4D temporal cardiac MRI cases. All data provide the segmentation annotations corresponding with the Left Ventricle (LV), Left Atrium (LA) and Myocardium (Myo) in the end-diastole (ED) and end-systole (ES) during the heartbeat cycle.

4.2 Implementation Details

Training. We trained the model using the Adam optimizer with betas equal to 0.9 and 0.99. The training batch size of the model was set to 1. We trained for a total of 1000 epochs with an initial learning rate of $5e^{-4}$ and decay by a factor of 0.5 in every 50 epochs. During training, for CardiacUDA [17] and CAMUS [18], we resized each frame to 384×384 and then randomly cropped them to 256×256. All frames were normalized to [0,1] during training. In temporal augmentation of datasets [17, 18], we randomly selected 32 frames from an echocardiogram video with a sampling ratio of either 1 or 2. For ACDC [19], we resampled all scans with a voxel spacing of 1.5 × 1.5 × 3.15mm and cropped them to $128 \times 128 \times 32$, normalized the intensity of all images to [-1, 1]. For spatial data augmentation of all datasets, we randomly applied flipping, rotation and Gaussian blurring. In CardiacUDA, we split the dataset into $8 : 2$ for training and validation. During testing, we reported results in 10 fully annotated videos. In the CAMUS [18] dataset, videos without annotation are used for only training, while we randomly split the remaining 450 annotated videos into 300/50/100 for training, validation and testing. In the ACDC [19], following the [11, 12], we split the training set in the ratio of 90 and 10 for training and testing. The reproduced methods strictly follow the official code and the description in the paper. For all experiments, We use Intel(R) Xeon(R) Platinum 8375C with $1\times$ RTX3090 for both training and inference. All reproduced methods strictly followed the training settings with their original paper in the same experimental environment.

Inference. For CardiacUDA and CAMUS, we resized videos to 384×384, cropped to 256×256 in central and normalized to [0,1]. We sample 32 frames that cover the segmentation annotation. When the sequence has more than 32 frames, the extra frames will be removed from the sequence, except for the first and the last one. The ACDC dataset remains the same sampling strategy as training in the inference stage, without any argumentation except for normalizing intensity to [-1,1].

Evaluation Metrics. For the evaluation of the quality of registered target frames, we follow [12] to use the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [48] to measure whether the Lagrangian motion field is accurately estimated between the first frame and the following wrapped frames. We also use the Dice [49] score to measure the discrepancy between tracked and ground-truth cardiac segmentation. For CardiacUDA [17], only the first frame and corresponding segmentation are provided for tracking the following 32 frames, then report the averaged results by the above metrics in these 32 frames. For CAMUS [18] and ACDC [19], frame and segmentation of the ED stage are used to track the go-after frames, and we report all the metrics in the ES stage. We evaluate diffeomorphic property by computing the percentage of non-positive values of the Jacobian determinant $det(J_\phi) \leq 0$ $(\%)$ on the Lagrangian motion field. In order to access the evaluation of comparing the physiological plausibility following the [50, 51], we also compute the mean absolute difference between the 1 and Jacobian determinant ($||J_\phi| - 1|$) over the tracking areas. For a fair comparison, we evaluate the computational efficiency and report the computational time in seconds (Times), the parameter quantities in millions (Params), and the tera-floating point operations per second (TFlops). *We also provide the result evaluated by Hussdorf Distance (HD) in Tables B1, B2 and B3 of Appendix Section B.*

https://doi.org/10.52202/079017-1098

Table 1: The performance[1] of different registration methods in Cardiac-UDA dataset [17]. Results were reported in structures (RV, RA, LV, LA) and the overall averaged Dice score (Avg. %).

2D Methods (256×256)	LV ↑	RV ↑	LA ↑	RA ↑	Avg. ↑	$\|\|J\|-1\|\downarrow$	$det(J_\phi)\leq 0\downarrow$	PSNR ↑	SSIM ↑	Times (s) ↓	Params (M) ↓	TFlops ↓
Non-rigid Registration												
LDDMM [6]	69.44±6.9	70.61±5.3	57.03±12	70.78±5.3	69.22±5.2	13.12±11.05	25.67±23.41	26.45±2.7	76.44±2.4	*177.9±2.3	-	-
RDMM [8]	70.50±7.3	71.12±6.3	57.10±12	72.22±6.0	70.84±6.3	5.102±1.067	8.602±6.350	26.80±2.6	76.92±1.9	*241.0±3.5	-	-
ANTs (SyN) [24]	73.51±6.6	74.12±5.7	60.49±14	74.69±4.6	73.71±5.8	16.09±8.031	40.06±28.56	27.96±2.4	76.52±2.5	*156.4±4.1	-	-
Deep Learning Based Registration												
VM-SSD [10]	74.26±8.3	74.85±5.2	66.78±18	76.24±7.4	75.86±4.2	0.374±0.021	0.262±0.305	29.01±2.5	75.89±1.8	0.011±0.0	0.118	0.010
VM-NCC [10]	74.04±7.2	76.20±5.9	67.54±14	77.36±4.2	76.51±4.2	0.685±0.052	0.905±1.229	28.53±2.5	75.77±2.3	0.011±0.0	0.118	0.010
SYMNet [36]	75.21±7.5	75.33±6.1	69.67±11	77.78±5.5	76.60±4.2	0.454±0.048	0.631±0.108	28.56±2.5	76.87±2.0	0.101±0.0	0.449	0.125
VM-DIF [9]	73.53±7.5	76.37±5.6	68.10±15	78.55±6.1	76.83±5.0	0.387±0.066	0.437±0.508	28.80±2.2	76.87±1.8	0.011±0.0	0.109	0.010
Ahn SS, et al. [31]	75.66±7.6	77.24±6.3	71.41±17	79.20±6.9	77.04±4.3	3.107±1.156	2.664±0.827	29.86±2.5	77.59±2.4	0.017±0.0	7.783	0.851
DiffuseMorph [12]	77.02±6.0	80.45±5.5	72.50±12	80.81±5.3	79.27±5.2	0.319±0.043	0.339±0.478	29.48±2.0	77.02±2.5	0.103±0.0	90.67	0.227
DeepTag [15, 16]	76.83±7.5	80.13±4.8	72.87±14	80.98±4.2	79.41±3.5	0.273±0.056	0.027±0.022	28.53±2.5	76.40±2.3	**0.011±0.0**	**0.107**	**0.010**
GPTrack-M (Ours)	76.94±7.6	81.72±6.4	73.13±16	80.85±6.4	81.64±2.8	0.286±0.069	0.119±0.084	31.28±2.0	78.22±2.4	0.013±0.0	0.467	0.015
GPTrack-L (Ours)	77.07±8.0	82.57±7.1	73.11±15	**81.24±6.4**	82.11±2.7	**0.250±0.044**	**0.019±0.017**	31.57±2.0	78.70±2.1	0.016±0.0	5.161	0.041
GPTrack-XL (Ours)	**78.51±7.9**	**82.48±6.0**	**73.43±12**	81.20±5.9	**82.37**±2.7	0.279±0.085	0.027±0.023	**32.03**±2.4	**80.04**±2.4	0.026±0.0	7.536	0.053

Table 2: The performance[1] of different registration methods in ACDC [19] dataset. Results reported in structures (RV, LV, Myo) and overall averaged Dice score (Avg. %).

3D Methods (128×128×32)	RV ↑	LV ↑	Myo ↑	Avg. ↑	$\|\|J\|-1\|\downarrow$	$det(J_\phi)\leq 0\downarrow$	PSNR ↑	SSIM ↑	Times (s) ↓	Params (M) ↓	TFlops ↓
Non-rigid Registration											
LDDMM [6]	73.61±8.5	65.62±8.5	56.44±13	72.39±18	451.8±162.3	653.5±371.2	31.20±3.8	84.59±6.0	*1533±8.4	-	-
RDMM [8]	76.43±7.8	69.50±9.1	62.19±14	75.51±12	144.2±63.67	266.0±165.3	31.66±3.9	84.36±5.4	*1715±26	-	-
ANTs (SyN) [24]	75.30±7.4	66.92±8.6	58.03±11	74.64±13	15.82±22.30	57.26±37.74	30.92±3.6	84.26±5.6	*1166±16	-	-
Deep Learning Based Registration											
VM-SSD [10]]	79.83±7.1	74.27±9.0	64.44±15	77.56±12	3.144±2.242	4.602±3.485	32.61±3.7	83.88±5.2	0.015±0.0	0.327	0.767
VM-NCC [10]	81.60±6.5	77.00±8.6	67.90±13	79.90±11	0.260±0.070	0.079±0.058	34.68±3.3	85.01±5.5	0.015±0.0	0.327	0.767
VM-DIF [9]	81.50±6.6	75.50±9.2	65.90±14	78.90±12	0.286±0.074	0.083±0.063	33.48±3.5	84.22±5.1	0.015±0.0	**0.327**	0.767
SYMNet [36]	80.46±6.4	77.81±9.4	66.22±14	79.47±13	0.341±0.062	0.121±0.054	32.91±3.5	83.55±4.9	0.414±0.0	1.124	0.226
NICE-Trans [52]	79.97±6.0	78.55±8.1	67.02±11	79.66±10	0.278±0.071	0.093±0.044	33.08±3.0	83.88±4.7	0.486±0.0	5.619	0.280
DiffuseMorph [12]	82.10±6.7	78.30±8.6	67.80±15	80.50±11	0.237±0.068	0.061±0.038	34.73±3.6	84.30±5.2	0.458±0.0	0.327	0.642
CorrMLP [53]	80.33±6.5	80.07±7.8	70.51±14	80.44±8.6	0.248±0.055	0.059±0.022	34.90±2.9	84.27±4.5	0.070±0.0	13.36	0.303
DeepTag [15, 16]	81.89±7.0	79.10±7.5	70.37±13	80.83±12	0.185±0.067	0.044±0.025	33.64±3.4	83.09±4.9	**0.015±0.0**	0.362	**0.113**
Transmatch [54]	81.22±7.0	80.34±6.8	71.21±12	81.35±9.8	0.226±0.050	0.077±0.054	33.89±3.3	84.78±4.9	0.325±0.0	70.71	0.603
FSDiffReg [11]	82.70±6.1	80.90±7.7	72.40±12	82.30±9.6	0.214±0.054	0.054±0.026	35.34±3.5	85.85±5.2	1.106±0.0	1.320	0.855
GPTrack-M (Ours)	81.65±7.0	80.77±7.5	71.53±16	81.45±10	0.209±0.081	0.047±0.035	34.82±3.2	85.78±5.3	0.022±0.0	0.418	0.201
GPTrack-L (Ours)	82.78±5.6	81.16±6.8	71.71±14	82.38±11	0.182±0.072	0.035±0.022	34.99±3.0	85.62±4.9	0.023±0.0	0.942	0.204
GPTrack-XL (Ours)	**82.91±5.8**	**81.23±8.2**	**72.86±9.0**	**82.65±10**	**0.178±0.024**	**0.032±0.021**	**35.52±3.1**	**86.19±5.0**	0.034±0.0	1.094	0.205

4.3 Results

Result of 3D Echocardiogram Video. In Table 1, we compare our method with state-of-the-arts 2D registration [9, 10, 12, 15, 31, 52] and non deep learning [6, 8, 24] methods in CardiacUDA dataset. In comparison to DeepTag [15], our GPTrack-XL reach 82.37% and 12.75 in DICE and HD scores with the best non-positive Jacobian determinant value, which denotes the learned motion field is smooth. The registration quality of our method also achieved the best with 32.03 (3.50 ↑) and 80.04 (3.64 ↑) in PSNR and SSIM compared to the second-best method, respectively. Though GPTrack introduces more learnable parameters and requires a small amount of additional computation, slightly increases inference time and TFlops compared to DeepTag [15] and VM-DIF [9]. GPTrack surpasses all other methods in the registration and verifies the necessity of formulating a strong temporal relationship among frames by using the recursive manner. Tables 1 and 3 show registration results of cardiac structures (LV, RV, LA, RA, Myo). Compared to other methods, our GPTrack can outperform existing baseline methods by a substantial margin. In areas such as the left atrium (LA) and left ventricular (LV), which usually cause larger deformation, our method can also provide better alignment than other approaches.

Table 3: The segmentation performance[1] of different cardiac structures in the CAMUS [18].

Methods (256×256)	CAMUS (Dice%)			
	LV	LA	Myo	Avg.
VM-NCC [10]	79.50±6.8	80.10±9.7	67.70±9.3	75.80±7.0
SYMNet [36]	85.24±7.0	82.77±9.6	76.36±6.2	81.84±4.5
VM-SSD [10]	86.30±6.7	85.20±9.3	77.90±6.9	83.10±4.5
Ahn SS, et al. [31]	86.42±7.2	83.95±8.9	75.68±8.4	82.33±4.1
DiffuseMorph [11]	85.76±5.9	84.49±8.7	76.65±6.3	83.57±4.5
VM-DIF [9]	87.70±6.0	85.40±10	**80.40±6.3**	84.50±3.7
DeepTag [15]	87.60±4.5	87.90±6.1	79.00±6.8	84.80±5.1
GPTrack-XL (Ours)	**88.63±4.6**	**89.13±8.0**	80.37±7.3	**85.29±4.2**

[1]Segmentation results are reported in the Dice score (%). **Bold**, underline denote the best results and the second best performance. The superscript * indicates computational time reported in only CPU implementation. We use the t-test for statistical significance analysis, where the p-value between the two methods is $p < 0.05$, indicating statistically significant improvements

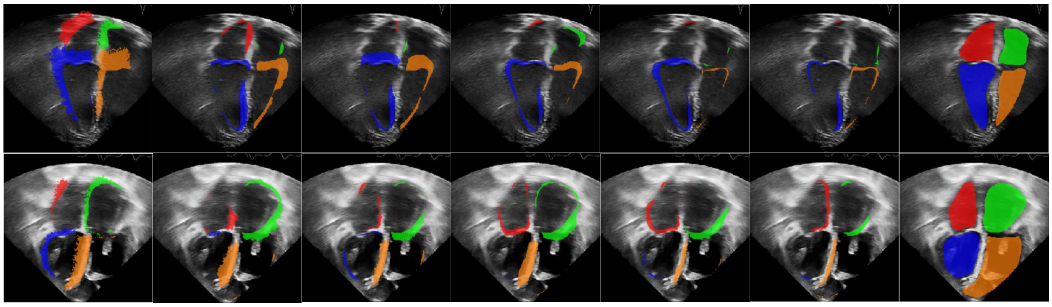

Figure 4: The visualization in 3D Echocardiogram video of motion tracking error. We visualised the last frame of tracking result and ground truth from 32 consecutive frames in CardiacUDA [17]. Colours Red, Blue, Green and Orange denote cardiac structures RA, RV, LV and LA, respectively.

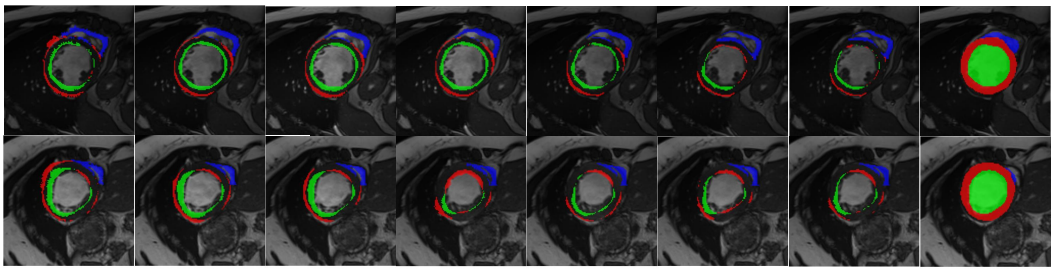

Figure 5: The visualization in 4D Cardiac MRI of motion tracking error. We visualised the result of the last frame tracking from ED to ES and corresponding ground truth in ACDC [17]. Colours Red, Blue, and Green denote cardiac structures MYO, LA, and LV, respectively.

As shown in Figure 4, the tracking error of our GPTrack and other methods show a significant difference in LA (Labelled by Orange Colour) and LV (Labelled by Green Colour) when compared to the ground truth. The methods [24, 10, 9, 31] present inaccuracy tracking results due to lack of the constraints on the consecutive motion and ignore the long-term temporal information. These results further verify that our specifically designed GPTrack for modelling motion patterns is more suitable for cardiac motion tracking on echocardiography.

Result of 4D Temporal Cardiac MRI Dataset. We compare our GPTrack method against with the state-of-the-art deep learning based methods [9, 10, 11, 12, 15] and different Non-rigid approaches [6, 8, 24]. As illustrated in Table 2, our GPTrack-XL achieves the best average DICE score of 82.65 compared to FSDiffReg [11] with 82.30. In registration quality, our GPTrack-XL reaches the highest scores, 31.52 and 86.19, in PSNR and SSIM, respectively. Moreover, our Jacobian determinant on deformation fields shows numbers comparable to other methods with the diffeomorphic constraint. All results are based on our fast and lightweight model, reducing around 96.93% inference time, 17.2% model parameters and 76.02% computational consumption (TFlops) compared to the second-best performance. In comparison to the diffusion-based method [12, 11], which requires enormous computation that hinders real-time inference and is nearly impossible to deploy in real scenarios, the GPTrack preserve light-weight and considerable performance by formulating cardiac motion patterns as the Gaussian process latent coding and bidirectionally understand the cardiac motion. The visualization result in Figure 5 also indicates our GPTrack can achieve better tracking accuracy.

4.4 Ablation Study

The Scale of Model Hyper-parameters. Table 4 shows the settings of the 2D/3D GPTrack-M/L/XL. For the 3D echocardiogram video dataset and 4D cardiac MRI dataset, the GPTrack with different scales has different patch sizes and dimension numbers. Referring to the result performed by Tables 3, 1 and 2, the registration result can be boosted by increasing the layers number and dimension size of GPTracks according to different requirements.

https://doi.org/10.52202/079017-1098

Table 4: The ablation study of the different configurations of our 2D / 3D GPTracks (M, L, XL).

Model	Layer	Patch Size	Dim	GFlops	Param (M)
GPTrack-M	2 / 2	16 / 8	64 / 32	1.54 / 20.1	0.467 / 0.418
GPTrack-L	2 / 2	16 / 8	256 / 64	4.18 / 20.4	5.161 / 0.942
GPTrack-XL	4 / 4	16 / 8	256 / 64	5.39 / 20.5	7.536 / 1.094

Table 5: Ablations of Bi-Directional (Bi-direct.) and Gaussian Process (GP) of **GPTrack-XL**.

Bi-direct.	GP	Dice.Avg	$det(J_\phi) \leq 0$
✗	✗	$79.83_{\pm 3.1}$	$0.048_{\pm 0.040}$
✗	✓	$80.81_{\pm 2.8}$	$0.052_{\pm 0.047}$
✓	✗	$81.21_{\pm 2.7}$	$0.039_{\pm 0.031}$
✓	✓	$\mathbf{82.37}_{\pm 3.2}$	$\mathbf{0.027}_{\pm 0.023}$

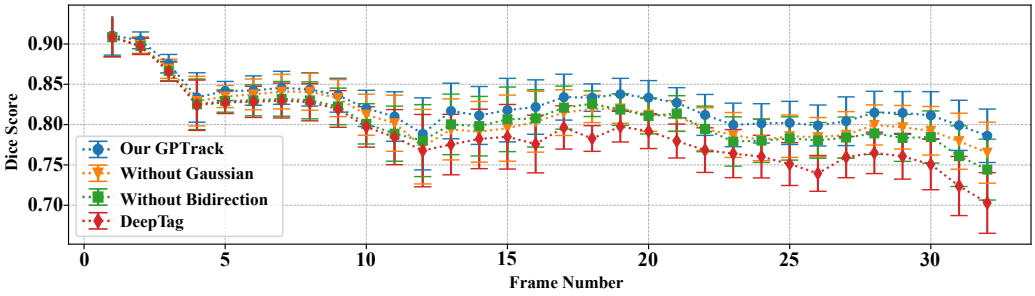

Figure 6: The Tracking Error performed by different methods in 32 consecutive frames of CardiacUDA [17] full annotated set.

The Ablation of Bidirectional Recursive Manner and Gaussian Process. Table 5 illustrates the Bi-directional layer in GPTracks can better formulate both forward and backward motion fields by aggregating the temporal information of the cardiac heartbeat cycle. The Gaussian Process models cardiac motion with strong prior knowledge from data, which makes more accurate predictions of the deformation field. The Figure 6 illustrate the tracking error from 1-st to 32-th frame in CardiacUDA [17] full annotated set. Our GPTrack and the DeepTag [15] both use the Lagrangian strain. However, the accuracy degrades significantly without aggregating the temporal information and GP when the input length increases. As shown in Figure 6, the tracking error after the 20-th frame becomes larger by using only Lagrangian strain, while introducing GP and bidirectional recursive methods can efficiently eliminate the tracking error by predicting more accurate motion fields.

5 Conclusion and Limitation

In this paper, we proposed a new framework named GPTrack to improve cardiac motion tracking accuracy. GPTrack innovatively aggregates both forward and backward temporal information by using the bidirectional recurrent transformer. Furthermore, we introduce the Gaussian Process to model the variability and predictability of cardiac motion. In experiments, our framework demonstrates the state-of-the-art in 3D echocardiogram and 4D cardiac MRI datasets. The limitation of our framework is that we use positional encoding as the prior knowledge of the cardiac motion, which may degrade the tracking performance in out-of-domain datasets. In our future work, we will build a more robust representation of cardiac motion and further our work across different medical domains. *We also provide the illustration of Broader Impacts, please see Section A2 in Appendix.*

Acknowledgements

This work was partially supported by grants from the National Natural Science Foundation of China (Grant No. 62306254), the Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone (Project No. HZQB-KCZYB-2020083), and the Research Grants Council of the Hong Kong Special Administrative Region, China (Project Reference Number: T45-401/22-N).

References

[1] Daniel Rueckert, Luke I Sonoda, Carmel Hayes, Derek LG Hill, Martin O Leach, and David J Hawkes. Nonrigid registration using free-form deformations: application to breast mr images. *IEEE transactions on medical imaging*, 18(8):712–721, 1999.

[2] J-P Thirion. Image matching as a diffusion process: an analogy with maxwell's demons. *Medical image analysis*, 2(3):243–260, 1998.

[3] Alessandro Becciu, Hans van Assen, Luc Florack, Sebastian Kozerke, Vivian Roode, and Bart M ter Haar Romeny. A multi-scale feature based optic flow method for 3d cardiac motion estimation. In *Scale Space and Variational Methods in Computer Vision: Second International Conference, SSVM 2009, Voss, Norway, June 1-5, 2009. Proceedings 2*, pages 588–599. Springer, 2009.

[4] Liang Wang, Patrick Clarysse, Zhengjun Liu, Bin Gao, Wanyu Liu, Pierre Croisille, and Philippe Delachartre. A gradient-based optical-flow cardiac motion estimation method for cine and tagged mr images. *Medical image analysis*, 57:136–148, 2019.

[5] KY Esther Leung, Mikhail G Danilouchkine, Marijn van Stralen, Nico de Jong, Antonius FW van der Steen, and Johan G Bosch. Left ventricular border tracking using cardiac motion models and optical flow. *Ultrasound in medicine & biology*, 37(4):605–616, 2011.

[6] 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*, 61:139–157, 2005.

[7] John Ashburner. A fast diffeomorphic image registration algorithm. *Neuroimage*, 38(1):95–113, 2007.

[8] Zhengyang Shen, François-Xavier Vialard, and Marc Niethammer. Region-specific diffeomorphic metric mapping. *Advances in Neural Information Processing Systems*, 32, 2019.

[9] Guha Balakrishnan, Amy Zhao, Mert R Sabuncu, John Guttag, and Adrian V Dalca. An unsupervised learning model for deformable medical image registration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9252–9260, 2018.

[10] Guha Balakrishnan, Amy Zhao, Mert R Sabuncu, John Guttag, and Adrian V Dalca. Voxelmorph: a learning framework for deformable medical image registration. *IEEE transactions on medical imaging*, 38(8):1788–1800, 2019.

[11] Yi Qin and Xiaomeng Li. Fsdiffreg: Feature-wise and score-wise diffusion-guided unsupervised deformable image registration for cardiac images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 655–665. Springer, 2023.

[12] Boah Kim, Inhwa Han, and Jong Chul Ye. Diffusemorph: Unsupervised deformable image registration using diffusion model. In *European conference on computer vision*, pages 347–364. Springer, 2022.

[13] Hanchao Yu, Shanhui Sun, Haichao Yu, Xiao Chen, Honghui Shi, Thomas S Huang, and Terrence Chen. Foal: Fast online adaptive learning for cardiac motion estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4313–4323, 2020.

[14] Nripesh Parajuli, Allen Lu, Kevinminh Ta, John Stendahl, Nabil Boutagy, Imran Alkhalil, Melissa Eberle, Geng-Shi Jeng, Maria Zontak, Matthew O'Donnell, et al. Flow network tracking for spatiotemporal and periodic point matching: Applied to cardiac motion analysis. *Medical image analysis*, 55:116–135, 2019.

[15] Meng Ye, Mikael Kanski, Dong Yang, Qi Chang, Zhennan Yan, Qiaoying Huang, Leon Axel, and Dimitris Metaxas. Deeptag: An unsupervised deep learning method for motion tracking on cardiac tagging magnetic resonance images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7261–7271, 2021.

[16] Meng Ye, Dong Yang, Qiaoying Huang, Mikael Kanski, Leon Axel, and Dimitris N Metaxas. Sequence-morph: A unified unsupervised learning framework for motion tracking on cardiac image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[17] Jiewen Yang, Xinpeng Ding, Ziyang Zheng, Xiaowei Xu, and Xiaomeng Li. Graphecho: Graph-driven unsupervised domain adaptation for echocardiogram video segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11878–11887, 2023.

[18] Sarah Leclerc, Erik Smistad, Joao Pedrosa, Andreas Østvik, Frederic Cervenansky, Florian Espinosa, Torvald Espeland, Erik Andreas Rye Berg, Pierre-Marc Jodoin, Thomas Grenier, et al. Deep learning for segmentation using an open large-scale dataset in 2d echocardiography. *IEEE transactions on medical imaging*, 38(9):2198–2210, 2019.

[19] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018.

 https://doi.org/10.52202/079017-1098

[20] Ruzena Bajcsy and Stane Kovačič. Multiresolution elastic matching. *Computer vision, graphics, and image processing*, 46(1):1–21, 1989.

[21] Christos Davatzikos. Spatial transformation and registration of brain images using elastically deformable models. *Computer Vision and Image Understanding*, 66(2):207–222, 1997.

[22] John Ashburner and Karl J Friston. Voxel-based morphometry—the methods. *Neuroimage*, 11(6):805–821, 2000.

[23] Nael F Osman, William S Kerwin, Elliot R McVeigh, and Jerry L Prince. Cardiac motion tracking using cine harmonic phase (harp) magnetic resonance imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 42(6):1048–1060, 1999.

[24] Brian B Avants, Charles L Epstein, Murray Grossman, and James C Gee. Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical image analysis*, 12(1):26–41, 2008.

[25] Julian Krebs, Tommaso Mansi, Hervé Delingette, Li Zhang, Florin C Ghesu, Shun Miao, Andreas K Maier, Nicholas Ayache, Rui Liao, and Ali Kamen. Robust non-rigid registration through agent-based action learning. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, pages 344–352. Springer, 2017.

[26] Xiaohuan Cao, Jianhua Yang, Jun Zhang, Dong Nie, Minjeong Kim, Qian Wang, and Dinggang Shen. Deformable image registration based on similarity-steered cnn regression. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, pages 300–308. Springer, 2017.

[27] Marc-Michel Rohé, Manasi Datar, Tobias Heimann, Maxime Sermesant, and Xavier Pennec. Svf-net: learning deformable image registration using shape matching. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, pages 266–274. Springer, 2017.

[28] Hessam Sokooti, Bob De Vos, Floris Berendsen, Boudewijn PF Lelieveldt, Ivana Išgum, and Marius Staring. Nonrigid image registration using multi-scale 3d convolutional neural networks. In *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, pages 232–239. Springer, 2017.

[29] Bob D De Vos, Floris F Berendsen, Max A Viergever, Marius Staring, and Ivana Išgum. End-to-end unsupervised deformable image registration with a convolutional neural network. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, pages 204–212. Springer, 2017.

[30] Kevinminh Ta, Shawn S Ahn, Allen Lu, John C Stendahl, Albert J Sinusas, and James S Duncan. A semi-supervised joint learning approach to left ventricular segmentation and motion tracking in echocardiography. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 1734–1737. IEEE, 2020.

[31] Shawn S Ahn, Kevinminh Ta, Allen Lu, John C Stendahl, Albert J Sinusas, and James S Duncan. Unsupervised motion tracking of left ventricle in echocardiography. In *Medical imaging 2020: Ultrasonic imaging and tomography*, volume 11319, pages 196–202. SPIE, 2020.

[32] Tony CW Mok and Albert CS Chung. Unsupervised deformable image registration with absent correspondences in pre-operative and post-recurrence brain tumor mri scans. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 25–35. Springer, 2022.

[33] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015.

[34] Xiaohuan Cao, Jianhua Yang, Jun Zhang, Qian Wang, Pew-Thian Yap, and Dinggang Shen. Deformable image registration using a cue-aware deep regression network. *IEEE Transactions on Biomedical Engineering*, 65(9):1900–1911, 2018.

[35] Adrian V Dalca, Guha Balakrishnan, John Guttag, and Mert R Sabuncu. Unsupervised learning for fast probabilistic diffeomorphic registration. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I*, pages 729–738. Springer, 2018.

[36] Tony CW Mok and Albert Chung. Fast symmetric diffeomorphic image registration with convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4644–4653, 2020.

[37] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015.

[38] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4571–4580, 2019.

[39] Junhwa Hur and Stefan Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5754–5763, 2019.

[40] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017.

[41] Jiewen Yang, Xingbo Dong, Liujun Liu, Chao Zhang, Jiajun Shen, and Dahai Yu. Recurring the transformer for video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14063–14073, 2022.

[42] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[43] Xingbo Dong, Jiewen Yang, Andrew Beng Jin Teoh, Dahai Yu, Xiaomeng Li, and Zhe Jin. Video-based face outline recognition. *Pattern Recognition*, 152:110482, 2024.

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[45] Simo Särkkä and Arno Solin. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

[46] Simo Särkkä, Arno Solin, and Jouni Hartikainen. Spatiotemporal learning via infinite-dimensional bayesian filtering and smoothing: A look at gaussian process regression through kalman filtering. *IEEE Signal Processing Magazine*, 30(4):51–61, 2013.

[47] Simo Sarkka and Jouni Hartikainen. Infinite-dimensional kalman filtering approach to spatio-temporal gaussian process regression. In *Artificial intelligence and statistics*, pages 993–1001. PMLR, 2012.

[48] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[49] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, 15:1–28, 2015.

[50] Chen Qin, Shuo Wang, Chen Chen, Huaqi Qiu, Wenjia Bai, and Daniel Rueckert. Biomechanics-informed neural networks for myocardial motion tracking in mri. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part III 23*, pages 296–306. Springer, 2020.

[51] Chen Qin, Shuo Wang, Chen Chen, Wenjia Bai, and Daniel Rueckert. Generative myocardial motion tracking via latent space exploration with biomechanics-informed prior. *Medical Image Analysis*, 83:102682, 2023.

[52] Mingyuan Meng, Lei Bi, Michael Fulham, Dagan Feng, and Jinman Kim. Non-iterative coarse-to-fine transformer networks for joint affine and deformable image registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 750–760. Springer, 2023.

[53] Mingyuan Meng, Dagan Feng, Lei Bi, and Jinman Kim. Correlation-aware coarse-to-fine mlps for deformable medical image registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9645–9654, 2024.

 https://doi.org/10.52202/079017-1098

[54] Zeyuan Chen, Yuanjie Zheng, and James C Gee. Transmatch: A transformer-based multilevel dual-stream feature matching network for unsupervised deformable image registration. *IEEE transactions on medical imaging*, 43(1):15–27, 2023.

[55] Noemi Carranza-Herrezuelo, Ana Bajo, Filip Sroubek, Cristina Santamarta, Gabriel Cristóbal, Andrés Santos, and María J Ledesma-Carbayo. Motion estimation of tagged cardiac magnetic resonance images using variational techniques. *Computerized Medical Imaging and Graphics*, 34(6):514–522, 2010.

[56] Simon Meister, Junhwa Hur, and Stefan Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[57] Xiaoyu Shi, Zhaoyang Huang, Dasong Li, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer++: Masked cost volume autoencoding for pretraining optical flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1599–1610, 2023.

[58] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. In *European conference on computer vision*, pages 668–685. Springer, 2022.

[59] Rico Jonschkowski, Austin Stone, Jonathan T Barron, Ariel Gordon, Kurt Konolige, and Anelia Angelova. What matters in unsupervised optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 557–572. Springer, 2020.

[60] Ziyang Zheng, Jiewen Yang, Xinpeng Ding, Xiaowei Xu, and Xiaomeng Li. Gl-fusion: Global-local fusion network for multi-view echocardiogram video segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 78–88. Springer, 2023.

[61] Jiewen Yang, Yiqun Lin, Bin Pu, Jiarong Guo, Xiaowei Xu, and Xiaomeng Li. Cardiacnet: Learning to reconstruct abnormalities for cardiac disease assessment from echocardiogram videos. In *European Conference on Computer Vision*. Springer, 2024.

[62] Bin Pu, Kenli Li, Jianguo Chen, Yuhuan Lu, Qing Zeng, Jiewen Yang, and Shengli Li. Hfsccd: a hybrid neural network for fetal standard cardiac cycle detection in ultrasound videos. *IEEE Journal of Biomedical and Health Informatics*, 2024.

Appendix

A1 Difference Between Optical Flow and Diffeomorphism Mapping

Optical flow (OF) based methods, often applied in object tracking of video sequences, have been explored by [3, 4, 5]. However, their effectiveness in medical imaging is limited due to challenges in accommodating large deformations and the inherent low quality of certain medical imaging modalities [4, 55], such as echocardiogram videos.

With the progression of deep learning, neural networks have also been employed to predict optical flow, which is crucial for predicting dynamic motion trajectories in video sequences. Notable implementations include FlowNet [37], iterative methods by [39], and self-supervised learning approaches by [38] and [56]. However, while supervised methods require a ground truth annotation for training cost functions [37, 39, 40, 57, 58], unsupervised approaches depend on photometric loss to ensure motion consistency [38, 56, 59], which can be challenging to obtain in medical images.

Last but not least, OF-based methods do not necessarily preserve topology, non-globally one-to-one (objective) smooth and continuous mapping with derivatives that are invertible. In cardiac motion tracking, we consider the deformation in each point of the adjacent frame to remain one-on-one mapping and be invertible for forward and backward deformation fields. Directly using the OF-based method to predict the motion field of cardiac motion may lead to incorrect estimation.

A2 Broader Impacts

Our work focuses on cardiac motion tracking with an unsupervised framework named GPTrack. The GPTrack framework has the potential to support medical imaging physicians, such as radiologists and sonographers, in observing the cardiac motion of patients. This is a fundamental task for assessing cardiac function, and we are able to provide decision support and improve analysis efficiency and analysis reproducibility in clinical scenarios.

Moreover, our new framework illustrates that cardiac motion can be formulated as strong prior knowledge, which is able to be utilised to enhance tracking accuracy. Also, our work presents several advantages that help us make progress towards these benefits, which improve the performance of automated motion field estimation algorithms. The method not only improves the precision of motion tracking and segmentation in both 3D and 4D medical image modailites [17, 18, 19, 60, 61, 62] but also provides a comprehensive observation of motion information to radiologists and sonographers to facilitate human assessment. However, this work may still remain gaps between real-world clinical utilization due to medical image analysis being a low failure tolerance application. The meaning of this work is to present a new direction for cardiac motion tracking, which is different from the conventional approach. In the current stage, the trained model in public datasets and the results presented are not specific to provide support for clinical use.

https://doi.org/10.52202/079017-1098

B Additional Quantitative Results and Visualization

We provide the experiment reported on Hussdorf Distance (HD) for datasets CardiacUDA [17], CAMUS [18] and ACDC [19] in Tables B1, B2 and B3, respectively. Furthermore, the consequent tracking results of 3D echocardiogram and 4D Cardiac MRI are presented in Figures B1 and B2.

Table B1: The performance of different registration methods in CardiacUDA [17]. Results report in Structures (LV, RV, LA, RV) and overall averaged Dice score (Avg. %). Segmentation results are reported in the Hussdorf Distance (HD). **Bold**, underline denote the best results and the second-best performance, respectively.

Methods (256×256)	CardiacUDA				
	LV	RV	LA	RA	Avg.
	Non-rigid Registration				
LDDMM [6]	18.12±3.9	17.33±3.2	16.94±3.4	16.77±3.6	17.27±3.0
RDMM [8]	17.47±3.2	17.69±3.4	16.36±3.2	15.97±2.9	17.01±2.3
ANTs (SyN) [24]	17.02±7.5	16.44±5.6	15.81±4.6	16.35±6.1	16.42±2.5
Deep Learning Based Registration					
VM-SSD [10]	17.09±3.3	16.11±2.6	15.60±2.8	15.46±3.2	16.11±2.4
VM-NCC [10]	16.84±2.9	15.78±3.1	15.93±3.4	15.16±2.3	15.88±2.5
SYMNet [36]	15.63±2.7	15.24±2.9	16.18±3.6	15.52±2.4	15.67±2.6
VM-DIF [9]	16.16±3.1	15.11±2.7	16.02±4.4	15.68±3.1	15.73±2.3
Ahn,S.S, et al. [31]	16.63±2.7	15.13±3.0	16.45±3.1	14.91±2.5	15.48±2.6
DiffuseMorph [12]	16.26±2.7	15.28±2.5	14.60±3.2	14.77±3.4	15.54±3.2
DeepTag [15]	15.81±2.8	15.68±1.9	14.39±2.6	**13.70±2.4**	14.94±2.4
GPTrack-M(Ours)	**14.63±2.6**	**14.77±3.0**	**12.19±2.5**	13.94±2.1	**13.87±2.3**

Table B2: The performance of different registration methods in CAMUS [18] dataset. Results report in Structures (LV, RV, Myo) and overall averaged Dice score (Avg. %). Segmentation results are reported in the Hussdorf Distance (HD). **Bold**, underline denote the best results and the second-best performance, respectively.

Methods (256×256)	CAMUS			
	LV	RV	Myo	Avg.
	Non-rigid Registration			
LDDMM [6]	7.210±3.8	10.65±7.7	6.592±2.3	7.305±2.5
RDMM [8]	6.307±3.4	9.584±6.7	6.911±2.5	6.831±2.6
ANTs (SyN) [24]	6.644±3.7	10.29±8.3	6.134±2.3	7.166±2.3
Deep Learning Based Registration				
VM-SSD [10]	5.769±3.0	8.755±7.6	5.231±1.9	6.240±1.8
SYMNet [36]	5.544±3.5	9.131±7.2	5.466±2.8	6.499±2.0
VM-NCC [10]	5.454±3.3	9.094±7.5	5.190±2.1	6.521±2.1
VM-DIF [9]	5.382±2.8	8.749±6.6	5.137±1.8	6.304±1.6
Ahn,S.S, et al. [31]	5.628±3.2	8.701±7.3	5.478±1.9	6.072±1.7
DiffuseMorph [12]	5.396±2.7	8.358±6.4	5.066±1.8	5.854±1.8
DeepTag [15]	5.207±3.1	7.651±6.1	**4.870±2.2**	5.388±1.6
GPTrack-M(Ours)	**4.722±2.8**	**6.857±5.7**	4.904±1.8	**4.945±1.1**

Table B3: The performance of different registration methods in ACDC [19] dataset. Results report in Structures (LV, RV, Myo) and overall averaged Dice score (Avg. %). Segmentation results are reported in the Hussdorf Distance (HD). **Bold**, underline denote the best results and the second-best performance, respectively.

Methods (256×256)	ACDC			
	LV	LA	Myo	Avg.
	Non-rigid Registration			
LDDMM [6]	5.817±2.4	6.662±2.9	6.337±2.8	6.562±2.1
RDMM [8]	5.430±2.5	6.268±2.4	5.953±2.2	5.728±1.5
ANTs (SyN) [24]	5.676±2.3	6.547±2.6	6.211±2.7	6.242±1.6
	Deep Learning Based Registration			
VM-SSD [10]	4.708±1.8	4.814±1.7	5.647±2.4	4.942±1.2
VM-NCC [10]	4.745±2.1	5.153±1.9	5.231±2.6	5.336±1.3
VM-DIF [9]	4.466±2.1	4.782±1.9	5.365±2.6	4.802±1.5
SYMNet [36]	4.864±2.3	5.149±2.1	5.552±2.7	5.254±1.9
NICE-Trans [52]	4.626±1.9	4.805±2.1	5.096±2.4	4.993±1.6
CorrMLP [53]	3.850±1.8	4.061±1.7	3.653±2.4	3.812±1.3
DiffuseMorph [12]	4.102±1.9	4.054±2.3	4.184±2.0	3.977±1.2
DeepTag [15, 16]	3.336±1.6	3.651±2.1	3.284±2.2	3.552±1.3
Transmatch [54]	3.904±1.9	3.855±2.1	3.770±1.9	3.716±1.4
GPTrack-M(Ours)	3.285±1.4	3.170±1.8	3.030±1.8	3.361±1.1
FSDiffReg [11]	**2.970**±1.3	3.298±2.0	2.862±1.8	3.283±1.2
GPTrack-XL(Ours)	3.147±1.5	**3.028**±1.9	**2.844**±1.8	**3.145**±1.1

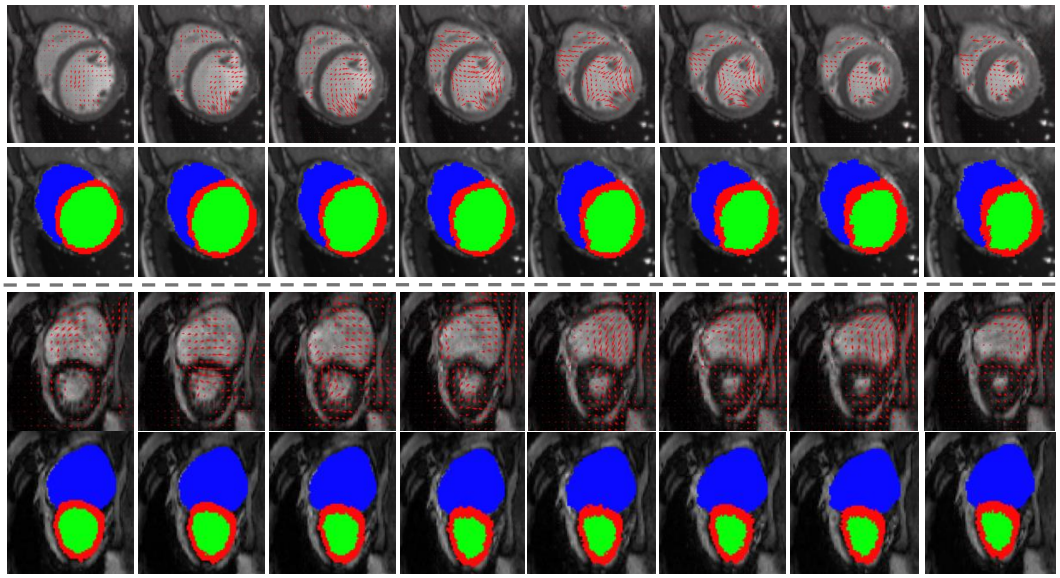

Figure B1: The visualization in 4D Cardiac MRI of estimated motion field and motion tracking results. We visualised the tracking result of the first frame (ED) to the last frame (ES) in ACDC [17]. Colours Red, Blue, and Green denote cardiac structures MYO, LA, and LV, respectively.

https://doi.org/10.52202/079017-1098

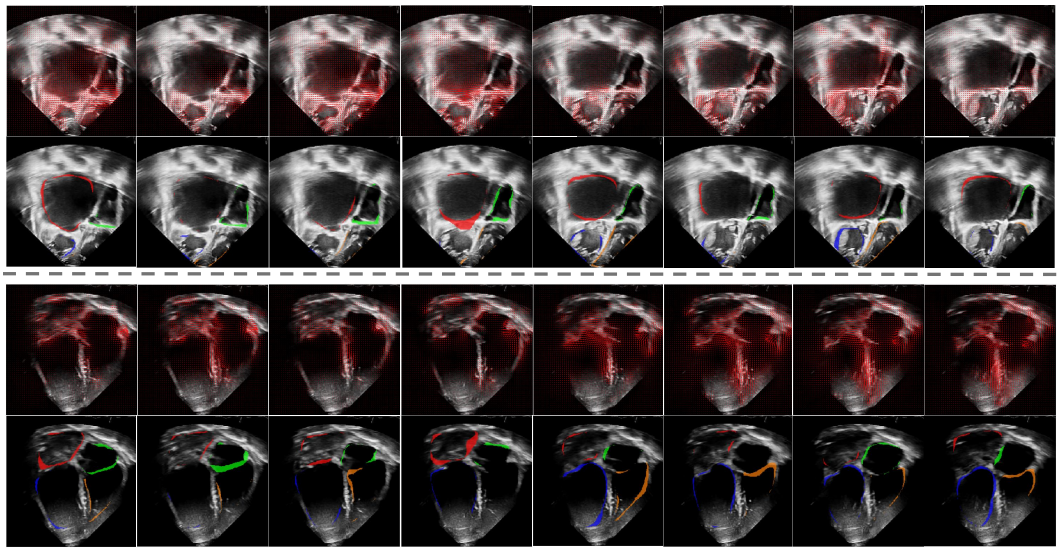

Figure B2: The visualization in 3D Echocardiogram video of estimated motion field and motion tracking error. We visualised tracking results from the first frame to the last frame, with ground truth from 8 consecutive frames in CardiacUDA [17]. Colours Red, Blue, Green and Orange denote cardiac structures RA, RV, LV and LA, respectively.

https://doi.org/10.52202/079017-1098

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Main claims are presented in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We already discuss the limiations in the last section.

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: We prove the full set of assumptions and a complete (and correct) proof in our paper.

https://doi.org/10.52202/079017-1098

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All the experimental results can be reproduced and we promise that all code will be made publicly.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All datasets in experiments are public datasets, all code will be made publicly.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have described all the training and testing details.

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Error bars, standard deviation and all details of result are presented in paper.

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

https://doi.org/10.52202/079017-1098

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please see the implementation details.

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We promise that all research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to Appendix A.2.

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No dataset or generative data.

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: For all datasets, we have acquired the license and permission for dataset usage.

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: N/A

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

https://doi.org/10.52202/079017-1098

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Not involve crowdsourcing nor research with human subjects.

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Not involve crowdsourcing nor research with human subjects.

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.