Improving Generalization by Learning Geometry-Dependent and Physics-Based Reconstruction of Image Sequences

Xiajun Jiang, Maryam Toloubidokhti, Jake Bergquist, Brian Zenger, Wilson W Good, Rob S MacLeod, *Member, IEEE,* and Linwei Wang, *Member, IEEE*

***Abstract*— Deep neural networks have shown promise in image reconstruction tasks, although often on the premise of large amounts of training data. In this paper, we present a new approach to exploit the geometry and physics underlying electrocardiographic imaging (ECGI) to learn efficiently with a relatively small dataset. We first introduce a non-Euclidean encoding-decoding network that allows us to describe the unknown and measurement variables over their respective geometrical domains. We then explicitly model the geometry-dependent physics in between the two domains via a bipartite graph over their graphical embeddings. We applied the resulting network to reconstruct electrical activity on the heart surface from body-surface potentials. In a series of generalization tasks with increasing difficulty, we demonstrated the improved ability of the network to generalize across geometrical changes underlying the data using less than 10% of training data and fewer variations of training geometry in comparison to its Euclidean alternatives. In both simulation and real-data experiments, we further demonstrated its ability to be quickly fine-tuned to new geometry using a modest amount of data.**

***Index Terms*— Geometric Deep Learning, Inverse Problems, Physics-Based Deep Learning.**

I. Introduction

Deep learning has shown state-of-the-art performance across a variety of image reconstruction tasks [1]–[6]. In some tasks, the imaging physics is partially known and modulated by specific parameters. For instance, heart or brain generates potentials that can be measured on the body or skull surface [7], [8]. This gives rise to (forward and inverse) mapping operators following the underlying quasi-static electromagnetic theory, but specific to the geometry on which the sources and measurements reside (*e.g.*, the heart and body surface).

In the context of Euclidean deep learning, one would attempt to learn such inverse mapping without the knowledge of the underlying geometry, such as the heart shape and relative position of between heart and torso [9]–[14]. This approach, as we will show, increases the need for training data and produce an inverse mapping not generalizable across geometries. It is possible to tackle the latter issue by using an information bottleneck to remove geometrical information from input data and thus make this inverse mapping invariant to geometrical factors [14]. Such an approach, unfortunately, requires even more training data to represent the variations arising from different geometries.

An interesting open question is thus whether learning such inverse mapping as a function of the underlying geometry would reduce the need of training data and improve the generalization of the learned function. Graph convolutional neural networks (GCNN) provide a promising approach to describe non-Euclidean variables defined over geometrical domains [15]. Significant efforts have been made in GCNN, such as node- and graph-level classifications, graph embedding, and graph generation [16]. However, to our knowledge, no previous works have reported learning inverse mappings between spatiotemporal variables defined on two separate graphs.

This paper presents a novel network to reconstruct non-Euclidean image sequences by directly learning the inverse mapping as a function of the underlying geometry of the problem. To describe the spatiotemporal variables (unknowns and measurements) over their respective geometrical domains, we first introduce an encoding-decoding architecture consisting of spatial-temporal graph convolutional neural networks (ST-GCNN) defined separately on each domain. To model the geometry-dependent physics in between, we then learn the inverse mapping as a function defined on a bipartite graph over the graphical embedding of these two geometrical domains with the functional form informed by the underlying physics. Extending on an earlier proof-of-concept [17], we focus on the generalization ability of this non-Euclidean image reconstruction network from two aspects. First, previous studies [14] based on Euclidean deep networks described that a stochastic formulation of the the same network, based on the theory of information bottleneck (IB) [18], could improve the generalization ability of the network by removing from the input data geometry information that are not present in the output solutions. By allowing the inverse mapping to change with the underlying geometry in the presented non-Euclidean network, we anticipate that this particular benefit of the IB

X. Jiang, M. Toloubidokhti, and L. Wang are with Golisano College of Computing and Information Sciences, Rochester Institute of Technology, Rochester, NY, 14623. (e-mail: xj7056@rit.edu, mt6129@rit.edu, and linwei.wang@rit.edu)

J. Bergquist, B. Zenger, W. W Good and R. S MacLeod are with University of Utah. (e-mail: jbergquist@sci.utah.edu, brian.zenger@hsc.utah.edu, wilsonwgood@gmail.com, and macleod@sci.utah.edu)

would be reduced. To test this conjecture, we develop the non-Euclidean network in both deterministic and stochastic formulations, and investigat their reconstruction performance differences. Second, we test the presented network in a series of generalization tasks with increasing difficulty, in comparison to Euclidean baselines with and without a geometry-invariant bottleneck [14], to test its ability to 1) reduce the diversity of the training data needed on the same geometry, 2) reduce the amount of geometrical variations needed in the training data, and 3) train across different geometry (*i.e.* different graphs), a process that is not possible with Euclidean learning unless these geometry are pre-registered. Finally, in *in-vivo* animal experiments, we demonstrate the ability of the presented network – after training on simulation data – to be quickly fine-tuned to a small amount of *in-vivo* data. All experiments are performaned in the application of reconstructing spatiotemporal electrical potentials on the ventricular surface from body-surface potentials (commonly known as electrocardiographic imaging (ECGI)) [19].

The main contributions of this paper include the following:

1) We present, to our knowledge, the first ST-GCNN approach to learn an inverse mapping between non-Euclidean variables that is geometry dependent and informed by the underlying imaging physics.
2) We present the first geometric deep learning approach to ECGI [19] that addresses the importance of geometry-specific mapping, which has been widely established in the ECGI literature [19], [20], but only rarely considered in emerging machine or deep learning solutions to ECGI.
3) Extending on a previous proof-of-concept, [17], we investigate the generalization ability of our method, both by examining its stochastic formulations based on the theory of IB, and by experimentation in a series of generalization tasks with increasing difficulty. We provid evidence for its ability to generalize without the stochastic formulation, to learn from a small amount of training data, as well as to learn and test across multiple different geometries.
4) We further extend our initial work [17] by investigating the ability of our method to be fine-tuned to a small amount of data on a new geometry.
5) We perform in-depth examinations into the effects of different neural network architecture designs, and introducing random edge dropping within the presented framework.

II. Background & Related Works

A. Electrocardiographic Imaging (ECGI)

Cardiac electrical sources produce time-varying voltage signals on the body surface, following quasi-static electromagnetism [21]. Given a pair of heart and torso geometries represented by their enclosing surfaces, the governing physics can be numerically approximated to relate signals on the heart $\mathbf{X}_t$ to those on the body surface $\mathbf{Y}_t$ [19]:

$$\mathbf{Y}_t = \mathbf{H}(\mathbf{g})\mathbf{X}_t \quad \forall t \in \{1, ..., T\}. \tag{1}$$

where $\mathbf{X}_t = [x_t(1), x_t(2), \ldots, x_t(M)]^T$ represents electrical potentials on M vertices of the heart mesh, and $\mathbf{Y}_t = [y_t(1), y_t(2), \ldots, y_t(N)]^T$ the electrical potentials on N vertices of the torso mesh, at time instant t. The forward operator $\mathbf{H}(\mathbf{g})$ defines the physics of their relationship, dependent on the given heart-torso geometry $\mathbf{g}$. Specifically, the signal on each torso vertex can be computed as a linear combination of signals on all heart vertices $y_t(i) = \sum_j x_t(j) \cdot h(g(i,j))$ for $i = 1, 2, \ldots N$ and $j = 1, 2, \ldots M$, where linear coefficients $h(g(i,j))$ are known to be inversely proportional to the relative distance between torso vertex i and heart vertex j [22], [23].

ECGI aims to reconstruct $\mathbf{X}_t$ from $\mathbf{Y}_t$ mathematically. In classic methods, imaging physics is incorporated as prior knowledge in the forward operator $\mathbf{H}(\mathbf{g})$. Numerical optimization and statistical inference methods are then used to seek the inverse solution that best fit the measurements under $\mathbf{H}(\mathbf{g})$, in combination with various constraints, such as the smoothness of the solution in space and time at different orders of derivatives [24], [25], its sparsity [26], [27], and *a priori* physiological knowledge about the spatiotemporal dynamics of the solution [7], [19], [28]. For instance, while the forward process is physically independent of time (quasi-static), various temporal constraints based on verified physiology have been considered useful for regularizing the inverse solutions [19], [28]. Overall, in this line of classic *physics-based* approaches, it is established that the accuracy of ECGI would rely on a forward operator that is specific to a subject's heart-torso geometry [20], [29].

With the success of modern machine learning and deep learning, data-driven approaches to ECGI have also emerged. Instead of incorporating imaging physics within a physics-based forward model, these approaches typically attempt to learn a direct inverse mapping using pairs of heart and body-surface data. For instance, the transmembrane potential and epicardial potential distributions are reconstructed from body surface potentials (BSPs) by a combination of clustering method and support vector regression (SVR) [10]. The correlation of body and heart surface potential in time was learned from the sequence of BSP data to iteratively reconstruct heart surface potentials in future timesteps [11]–[13].

Till now, few data-driven ECGI approaches considered the fact that the inverse mapping should be specific to the underlying geometry. As a result, the learned inverse mapping—as in the examples provided above—has to be restricted to the same geometry on which the training was performed. This largely limits the clinical value of these approaches to be applied across patients. To address this challenge, investigators have performed the learning of the inverse mapping between BSPs and activation maps offline and transferred the results onto patient specific anatomies to achieve fast personalized predictions online [30]. A similar approach is to learn an inverse mapping that is invariant to geometry by removing geometry-related information from the input ECG data using an information bottleneck [14]. However, this approach requires additional training data that represents variations from different geometries. Alternatively, the geometry can be incorporated by conditioning the reconstructions of electrical potentials on 2D image scans of the heart shape [9]. It is not clear how to extend this approach to consider the most important geometrical factors in ECGI—the relative position between the heart and torso.

Our ST-GCNN approach departed from previously reported techniques to directly learn the inverse mapping as a function of geometry by 1) modeling the inverse mapping over non-Euclidean geometry and 2) incorporating physics relationship between the measurement and unknown into the design of the inverse mapping at the latent space.

B. Graph Convolutional Neural Networks (GCNN)

GCNN provides an appealing method to learn functions on non-Euclidean variables defined over geometrical domains [15]. To extend the convolution operator to graph-structured data, the spectral approach utilizes a graph Fourier transformation that projects the input graph signal to the orthonormal space whose basis is formed by eigenvectors of the normalized graph Laplacian [16]. These approaches are typically limited to a single graph because the spectral filter coefficients depend on the selected basis [15]. The spatial approach defines graph convolutions based on the spatial relation of each vertex [16]. For instance, in SplineCNN [31], the graph convolution operator integrates the signals within the vertex neighborhood, where the weighting function is a B-spline that considers the relative position between a vertex and its neighbor. In this study, we adopted the SplineCNN as the building block of the ST-GCNN model, since the spline convolution filter can be applied to different graphs.

Spatial-temporal graph neural networks (ST-GCNNs) capture spatial and temporal dependencies of a graph simultaneously [16]. Recurrent neural network (RNN)-based approaches use a recurrent unit to pass hidden states and filter them together with inputs using graph convolutions [32]. The major drawback of these approaches is that they suffer from time-consuming propagation and gradient explosion/vanishing problems [16]. Convolutional neural network (CNN)-based approaches mitigate this drawback by interleaving 1D CNN layers with graph convolutional layers to learn temporal and spatial dependencies respectively [33]. Our version of ST-GCNN applied to ECGI is inspired by the latter structure.

To learn graph embeddings, graph autoencoders have been investigated to reconstruct, or to generate new graphs [34]–[36]. This line of work mainly concentrates on reconstructing or generating new structures of the graphs. In contrary, the presented work utilizes graph encoding and decoding for the purpose of incorporating structural information from given geometry, and for learning a geometry-dependent mapping between signals on the input and output graphs.

III. Methodology

To respect the geometry-dependent physics behind the problem, our method learns a geometry-dependent inverse mapping by 1) describing $\mathbf{X}_t$ and $\mathbf{Y}_t$ in their respective geometrical domains, and 2) explicitly modeling their relationship at the latent space as a function of the geometry. We realize our method in an encoder-decoder architecture with ST-GCNNs as summarized in Fig. 1: a ST-GCNN encoder embeds $\mathbf{Y}_t$ over the torso geometry, and a ST-GCNN decoder generates $\mathbf{X}_t$ over the ventricular geometry; at the latent space, the relationship between the latent variables of $\mathbf{Y}_t$ and $\mathbf{X}_t$ – as informed by the actual imaging physics – is assumed to be linear with coefficients as a function over the graph embedding of the two geometries. Following past ECGI works that showed the importance of including the temporal dimension into the reconstruction, we consider reconstructing the spatiotemporal signals on the heart over time.

A. Encoding-Decoding with ST-GCNNs

As $\mathbf{X}_t$ and $\mathbf{Y}_t$ are temporal sequences that exist within a 3D geometry, we describe their generation/embedding with ST-GCNNs that consist of interlaced graph convolution in space and regular compression in time as illustrated in Fig. 1.

1) Geometrical Representation in Graphs: Triangular meshes of the heart and torso surfaces are represented as two separate undirected graphs: $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{U}, \mathbf{F})$, where vertices $\mathcal{V}$ consist of all V mesh nodes and edges $\mathcal{E}$ describe the vertex connection as defined by the triangular mesh. $\mathbf{U} \in [0,1]^{V\times V\times 3}$ consists of edge attributes $\mathbf{u}(i,j)$ between vertex i and j as normalized differences in their 3D coordinates if an edge exists. $\mathbf{F} \in \mathbb{R}^{V\times M\times T}$ represents the time sequences of features across all vertices, where the feature represents the signals at the input/output level when $M = 1$ and feature maps in the middle layers, where M is the size of the feature dimension.

During encoding and decoding, we apply hierarchical graph representations that coarsens as it gets closer to the bottleneck to the two geometries. In contrast to general graphs, this hierarchical representation should satisfy a constraint that the topology of the geometry must be preserved in its hierarchical representations to prevent non-physical spatial propagation of signals. The hierarchical representations are obtained by a specialized mesh coarsening method in the Computational Geometry Algorithms Library (CGAL) [37], defined prior to the training of the ST-GCNN.

2) Spatial Graph Convolution: We use a continuous spline kernel for spatial convolution such that it can be applied across different graphs [31]. For each channel of the feature map at each time instant, the convolution kernel is defined as:

$$g_l(\mathbf{u}) = \sum_{\mathbf{p}\in\mathcal{P}} w_{\mathbf{p},l} B_{\mathbf{p}}(\mathbf{u}), \tag{2}$$

where $1 \le l \le C$ and C is the number of channels. The spline basis $B_{\mathbf{p}}(\mathbf{u}) = \prod_{r=1}^{d} N^m_{r,p_r}(\mathbf{u})$ with N^m_{r,p_r} denoting d, an open B-spline basis of degree m based on equidistant knot vectors, $\mathcal{P} = (N^m_{1,r})_r \times \ldots \times (N^m_{d,r})_r$ is the Cartesian product of the B-spline bases, and $w_{\mathbf{p},l}$ are trainable parameters.

Given kernel $\mathbf{g} = (g_1, \ldots, g_C)$ and graph node features $\mathbf{f} \in \mathbb{R}^{V\times M}$ at each time instant, spatial convolution for vertex $i \in \mathcal{V}$ with its neighborhood $N(i)$ is defined as:

$$(f_l * g_l)(i) = \sum_{j\in N(i), \mathbf{p}\in\mathcal{P}(\mathbf{u}(i,j))} f_l(j) \cdot g_l(\mathbf{u}(i,j)). \tag{3}$$

Since the B-spline basis in the (2) is conditioned on local geometry, the learned kernel can be applied across graphs and the convolution incorporates geometrical information within the graph. This spatial convolution is independently applied to each time instant of the signal sequence in parallel.

This article has been accepted for publication in IEEE Transactions on Medical Imaging. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMI.2022.3218170

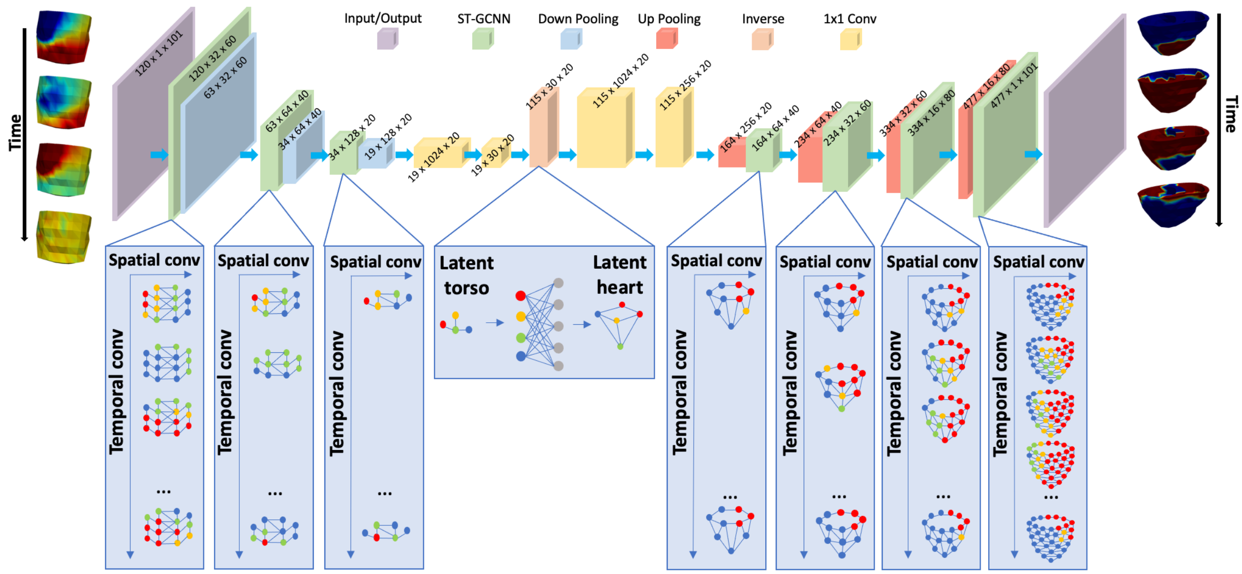

Fig. 1. Outline of the ST-GCNN inverse imaging network. The size of the feature maps follows the format of #Vertices×#Feature×#Time.

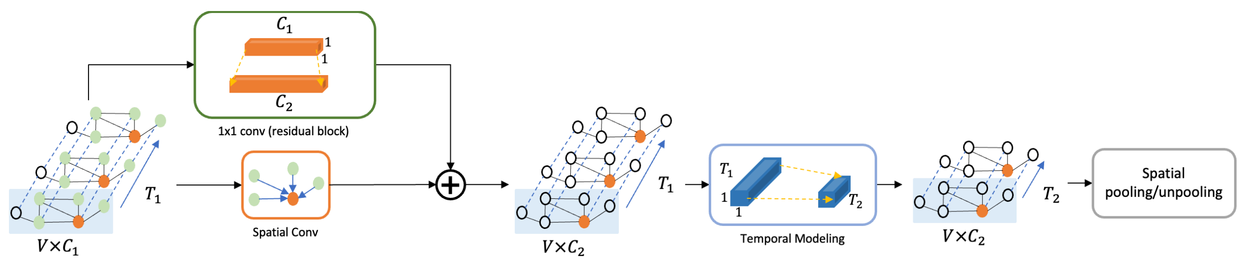

Fig. 2. Illustration of the structure of one ST-GCNN layer followed by a spatial pooling/unpooling layer from Fig. 1. C_1 and C_2 are the number of channels before/after the spatial/regular convolution. T_1 and T_2 are the temporal dimensions before/after the temporal convolution.

To make the network deeper and more expressive in feature representation, residual blocks are introduced to pass the input of spatial convolution through a skip connection with 1D convolution. Fig. 2 illustrates all components in an ST-GCNN layer.

3) Temporal Modeling: The spatial convolution is interlaced with temporal modeling. Common approaches for temporal modeling include using a standard convolutional kernel [38] or a recurrent unit of time sequences [39]. Here, we apply directly the fully connected layers on time sequences, where the parameter of the layer is shared across each vertex and feature. The size of the output of the fully connected layer is set to compress the time sequence in dimension in the encoder, while expanding in the decoder. The geometry graph remains the same for the complete temporal sequences. In Section IV-F, this temporal modeling will be compared with alternative RNN or CNN options in experimental evaluations.

4) Pooling and Unpooling: Pooling and unpooling in space are carried out on the hierarchical graph representation of the two geometries described in Section III-A.1. Using $\mathcal{G}_o$ to denote a graph with N_1 vertices and $\mathcal{G}_c$ its coarsened graph with N_2 vertices, we use a binary matrix $\mathbf{P} \in \mathbb{R}^{N_1 \times N_2}$, where $\mathbf{P}_{ij} = 1$ if vertex i in $\mathcal{G}_o$ is grouped to vertex j in $\mathcal{G}_c$, and $\mathbf{P}_{ij} = 0$ otherwise. Here, each vertex on $\mathcal{G}_o$ is grouped to its nearest vertex on $\mathcal{G}_c$. Given a feature map $\mathbf{f}_o \in \mathbb{R}^{N_1 \times M}$ over $\mathcal{G}_o$ and $\mathbf{f}_c \in \mathbb{R}^{N_2 \times M}$ over $\mathcal{G}_c$, the pooling operation is defined by $\mathbf{f}_c = \mathbf{P}_n^T \mathbf{f}_o$ and the unpooling operation is defined by $\mathbf{f}_o = \mathbf{P}\mathbf{f}_c$, where $\mathbf{P}_n^T$ is column normalized from $\mathbf{P}$.

5) Summary: As summarized in Fig. 2, each ST-GCNN block consists of spatial graph convolution, temporal compression, and spatial pooling/unpooling, as described above. Denoting the latent features of the body-surface signals as $\mathbf{Z}_b$ and those of the heart-surface signals as $\mathbf{Z}_h$, respectively, we can represent the encoder and decoder as $\mathbf{Z}_b = E_\theta(\mathbf{Y})$ and $\hat{\mathbf{X}} = D_\phi(\mathbf{Z}_h)$, where θ and ϕ are parameters of the encoder and decoder, respectively.

B. Learning Latent Geometry-Dependent Physics

As explained in (1), $\mathbf{Y}_t$ on one torso vertex can be represented as a linear combination of $\mathbf{X}_t$ from all heart vertices, where the coefficients are determined by the relative position between each pair of torso-heart vertices. We assume this linearity to hold between $\mathbf{Z}_h$ and $\mathbf{Z}_b$ in the latent space. Specifically, for $\mathbf{z}_h(i)$ on vertex i of the latent heart mesh, we define it as a linear combination of latent features $\mathbf{z}_b(j)$ across all vertices j of the latent torso mesh.

One option to learn this linear mapping between $\mathbf{Z}_b$ and $\mathbf{Z}_h$ is a fully connected layer. However, the resulting learned relationship will not consider the underlying geometry and, more importantly, will not be applicable to different heart-torso geometries with different numbers of graph vertices. Instead,

This article has been accepted for publication in IEEE Transactions on Medical Imaging. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMI.2022.3218170

we explicitly model the learnable coefficients as functions of the relative position between embedded heart and torso geometries. To do so, we construct a bipartite graph such that an edge with attribute $\mathbf{u}(i,j)$ exists between each pair of heart and torso vertices in their graph embeddings. We then define $\mathbf{z}_h(i)$ as a linear combination of $\mathbf{z}_b(j)$ across all vertices j, where linear coefficients $\hat{\mathbf{h}}$ are learnable as a function over $\mathbf{u}(i,j)$:

$$\mathbf{z}_h(i) = \sum_j \mathbf{z}_b(j) \cdot \hat{\mathbf{h}}(\mathbf{u}(i,j)). \tag{4}$$

Exploiting the similarity between (4) and (3), we describe (4) using spline convolution with the geometry-dependent coefficients $\hat{\mathbf{h}}$ learned as the spline convolution kernel. We denote the geometry-dependent inverse function as $\mathbf{Z}_h = h_\rho(\mathbf{Z}_b)$ with network parameter ρ. Aside from being physics-informed, this geometric parameterization allows the learned function to generalize across different geometries.

C. Deterministic and Stochastic Formulations

As explained earlier, we developed both a deterministic and stochastic formulation of the non-Euclidean encoding-decoding networks, in order to investigate whether the use of IB theory is still necessary for improving the ability of the network to generalize to different geometries.

For the deterministic model, its parameters θ, ρ and ϕ are optimized by minimizing the mean square error between the reconstructed $\hat{\mathbf{X}}^{(i)}$ on training data $\{\mathbf{X}^{(i)}, \mathbf{Y}^{(i)}\}_{i=1}^N$:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} ||\mathbf{X}^{(i)} - D_\phi\left(h_\rho\left(E_\theta\left(\mathbf{Y}^{(i)}\right)\right)\right)||_2^2. \tag{5}$$

For the stochastic model, $\mathbf{Z}_b$ is modeled with a Gaussian distribution whose mean and variance are obtained by neural networks: $p_\theta(\mathbf{Z}_b|\mathbf{Y}) = \mathcal{N}(\mathbf{Z}_b|\boldsymbol{\mu}_b(\mathbf{Y}), \boldsymbol{\sigma}_b^2(\mathbf{Y}))$. We apply reparameterization $\mathbf{Z}_b = \boldsymbol{\mu}_b + \boldsymbol{\sigma}_b \odot \boldsymbol{\epsilon}$ as described in [40], where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$ and $\odot$ is Hadamard product. We draw a random sample from the distribution of $\mathbf{Z}_b$, and then apply $\mathbf{Z}_h = h_\rho(\mathbf{Z}_b)$ to obtain the sample for $\mathbf{Z}_h$. The decoder then reconstructs from this sample. From the theory of information bottleneck [18], we minimize:

$$loss_{IB} = -I(\mathbf{X};\mathbf{Z}_h) + \beta I(\mathbf{Z}_b;\mathbf{Y}) \tag{6}$$

where $I(\mathbf{X};\mathbf{Z}_h)$ is the mutual information between the output and latent features of the heart signals, $I(\mathbf{Z}_b;\mathbf{Y})$ is the mutual information between the measurement and latent features of the torso signals, and β is the multiplier of the KL-divergence term. For the first term in (6) we have:

$$\begin{aligned}
I(\mathbf{X};\mathbf{Z}_h) &= \int p(\mathbf{X},\mathbf{Z}_h)\log\frac{p(\mathbf{X}|\mathbf{Z}_h)}{p(\mathbf{X})}d\mathbf{X}d\mathbf{Z}_h \\
&= H(\mathbf{X}) + \int p(\mathbf{X},\mathbf{Z}_h)\log p(\mathbf{X}|\mathbf{Z}_h)d\mathbf{X}d\mathbf{Z}_h \\
&= H(\mathbf{X}) + \int p(\mathbf{Z}_h)p(\mathbf{X}|\mathbf{Z}_h)\log\frac{p(\mathbf{X}|\mathbf{Z}_h)}{q(\mathbf{X}|\mathbf{Z}_h)}d\mathbf{X}d\mathbf{Z}_h \\
&+ \int p(\mathbf{Z}_h)p(\mathbf{X}|\mathbf{Z}_h)\log q(\mathbf{X}|\mathbf{Z}_h)d\mathbf{X}d\mathbf{Z}_h \\
&= H(\mathbf{X}) + D_{KL}(p(\mathbf{X}|\mathbf{Z}_h)||q(\mathbf{X}|\mathbf{Z}_h)) \\
&+ \int p(\mathbf{X},\mathbf{Y},\mathbf{Z}_h)\log q(\mathbf{X}|\mathbf{Z}_h)d\mathbf{X}d\mathbf{Y}d\mathbf{Z}_h \\
&\geq E_{p(\mathbf{X},\mathbf{Y})}[E_{p(\mathbf{Z}_b|\mathbf{Y})}[\log q(\mathbf{X}|\mathbf{Z}_h)]],
\end{aligned}$$

where $\mathbf{Z}_h = \hat{\mathbf{H}}(\mathbf{g})\mathbf{Z}_b$ and $H(\mathbf{X}) = \int p(\mathbf{X})\log p(\mathbf{X})d\mathbf{X}$. We set $q_\phi(\mathbf{X}|\mathbf{Z}_h)$ to be a Gaussian distribution parameterized by the decoder: $q_\phi(\mathbf{X}|\mathbf{Z}_h) = \mathcal{N}(\mathbf{X}|\mu_h, \sigma_h^2)$.

For the second term in (6):

$$\begin{aligned}
I(\mathbf{Z}_b;\mathbf{Y}) &= \int p(\mathbf{Y},\mathbf{Z}_b)\log\frac{p(\mathbf{Z}_b|\mathbf{Y})}{p(\mathbf{Z}_b)}d\mathbf{Y}d\mathbf{Z}_b \\
&= \int p(\mathbf{Y},\mathbf{Z}_b)\log\frac{p(\mathbf{Z}_b|\mathbf{Y})r(\mathbf{Z}_b)}{r(\mathbf{Z}_b)p(\mathbf{Z}_b)}d\mathbf{Y}d\mathbf{Z}_b \\
&= \int p(\mathbf{Y})p(\mathbf{Z}_b|\mathbf{Y})\log\frac{p(\mathbf{Z}_b|\mathbf{Y})}{r(\mathbf{Z}_b)}d\mathbf{Y}d\mathbf{Z}_b \\
&- D_{KL}(p(\mathbf{Z}_b)||r(\mathbf{Z}_b)) \\
&\leq \int p(\mathbf{X},\mathbf{Y})p(\mathbf{Z}_b|\mathbf{Y})\log\frac{p(\mathbf{Z}_b|\mathbf{Y})}{r(\mathbf{Z}_b)}d\mathbf{X}d\mathbf{Y}d\mathbf{Z}_b \\
&= E_{p(\mathbf{X},\mathbf{Y})}[D_{KL}(p(\mathbf{Z}_b|\mathbf{Y})||r(\mathbf{Z}_b))],
\end{aligned}$$

where the KL divergence is analytically available when $r(\mathbf{Z}_b)$ is set to be a standard Gaussian distribution: $r(\mathbf{Z}_b) = \mathcal{N}(\mathbf{z}_b|\mathbf{0},\mathbf{I})$ and the latent distribution of torso graph embedding as a Gaussian distribution: $p_\theta(\mathbf{Z}_b|\mathbf{Y}) = \mathcal{N}(\mathbf{Z}_b|\boldsymbol{\mu}_b(\mathbf{Y}))$. Therefore, we have:

$$\begin{aligned}
loss_{IB} &= -I(\mathbf{X};\mathbf{Z}_h) + \beta I(\mathbf{Z}_b;\mathbf{Y}) \\
&\leq E_{p(\mathbf{X},\mathbf{Y})}[-E_{p_\theta(\mathbf{Z}_b|\mathbf{Y})}[\log q_\phi(\mathbf{X}|\mathbf{Z}_h)]] \\
&+ \beta D_{KL}(p_\theta(\mathbf{Z}_b|\mathbf{Y})||r(\mathbf{Z}_b)) = \mathcal{L}_{IB}.
\end{aligned}$$

IV. Simulation Data Experiments

In controlled simulation experiments, we evaluated the performance of the reconstruction network in a series of generalization tasks, with increasing difficulty in terms of how close the test geometry is to those in training data. We further assessed how the performance of the network changed as the diversity of the training data decreased. We compared the performance of our network to that of a Euclidean encoding-decoding networks, as described by Ghimire *et al.* [14] in a deterministic formulation as well as a stochastic formulation with a geometry-invariant bottleneck. In a subset of experiments where Euclidean networks do not apply (Section IV-D, training across multpile geometry), we further compared the performance of our network to classic ECGI approaches utilizing known physics-based forward operators.

This article has been accepted for publication in IEEE Transactions on Medical Imaging. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMI.2022.3218170

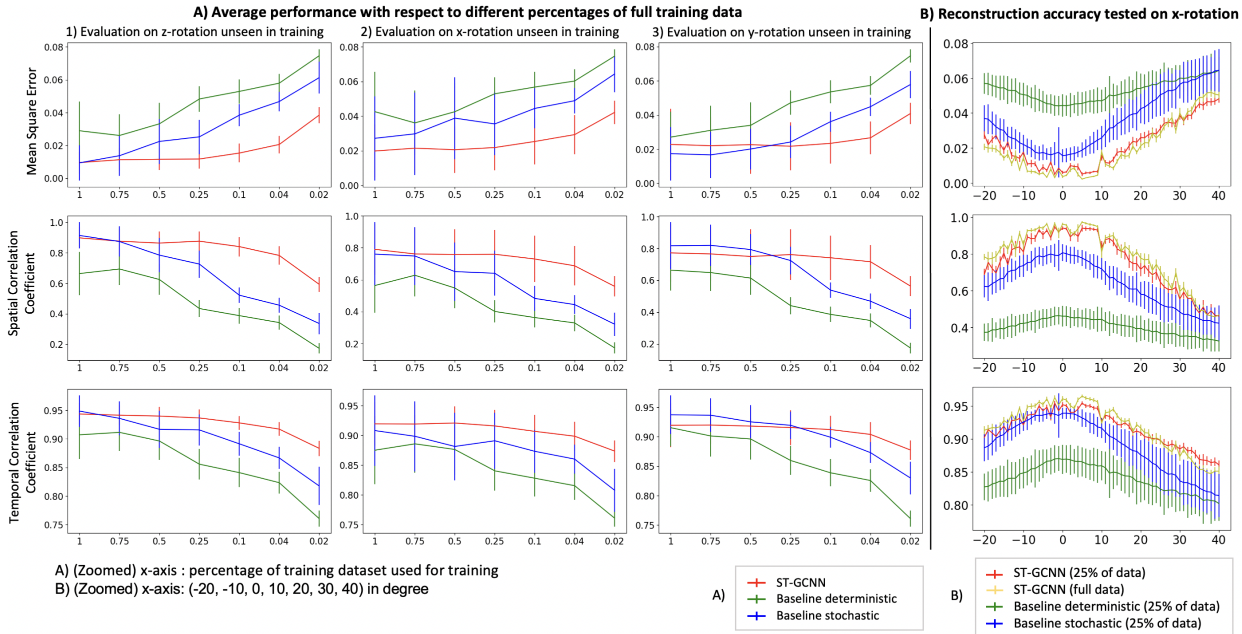

Fig. 3. Summary of average performance with respect to different percentages of full training data among the three comparison models.

A. Models, Data, and Training

Our presented network consisted of three ST-GCNN blocks and two standard convolutional layers in the encoder, one spline convolutional layer in the latent inverse mapping, and four ST-GCNN blocks and two standard convolutional layers in the decoder, as detailed in Fig. 1. We used B-spline basis degree of $m = 1$ with kernel size of $k_1 = k_2 = k_3 = 3$ in all graph convolution layers. The spatial and temporal dimensions in each level of the encoder were $[120, 63, 34, 19]$ and $[101, 60, 40, 20]$, respectively; and $[115, 164, 234, 334, 477]$ and $[20, 40, 60, 80, 101]$ in the decoder, respectively. We used ELU activation [41], an ADAM optimizer [42], and a learning rate of 5×10^{-4}. The Euclidean baselines followed the architectures presented in [14] consisting of cascaded LSTMs and fully-connected layers.

For training, we generated pairs of simulated potentials on a heart-torso mesh. We simulated realistic spatiotemporal propagation sequences of action potentials by the Aliev-Panfilov model [43], considering a combination of 38 origins of activation and 16 spatial distributions of scar tissue (totaling 531 data samples for a single geometry). We then rotated the heart by -2° to 2° around the z-axis, obtaining approximately 2700 sets of body-surface potentials. All body-surface potentials were corrupted with 20 dB SNR of Gaussian noises before performing inverse imaging.

Testing data were generated in a similar fashion, with additional geometry changes as detailed below. The reconstruction accuracy was measured by the mean square error (MSE), spatial correlation coefficient (SCC), and temporal correlation coefficient (TCC) between the reconstructed and actual potential sequence on the heart surface. While MSE measures the quantitative errors of the reconstructed signals on the heart, the SCC and TCC measures the correlation between the reconstructed and reference signals. We considered the correlation both for spatial signals at each time instance (SCC) and temporal signals at each location of the heart (TCC).

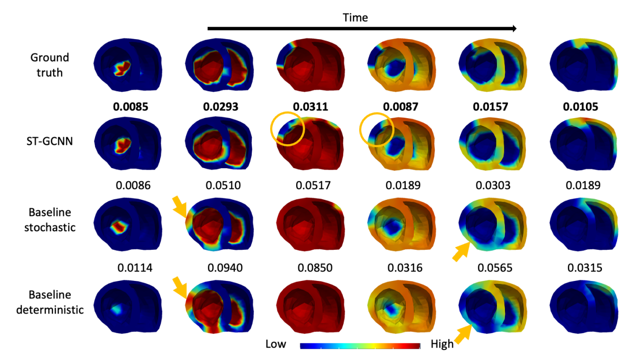

Fig. 4. Examples of reconstructed electrical activity trained on 25% of full training data, tested at rotation x = $\mathbf{10°}$. The MSE value is shown for each model at each timestep. Both of the two baseline models showed high errors on the free wall of the left ventricle during depolarization/repolarization (arrows), while our presented model predicted the most accurate signal propagation pattern and the scar location (circles).

B. Generalization to Unseen Heart Rotations

In this set of experiments, we first applied the trained models to 21,771 sets of body-surface potentials generated by rotating the heart by -20° to 20° around the z-axis, a range far outside that considered in training. We then tested the trained models on a different set of 64,782 body-surface potentials generated from novel heart rotations around the x-axis (-20° to +40°) and y-axis (-20° to +40°), types of rotations not seen in training. In both experiments, we examined the change of performance of the trained models when 1) we randomly sampled the training data with respect to the combinations of sites of activation and scar locations by a rate of 2%, 4%, 10%, 25%, 50%, 75%, and 100% of the complete training data, and 2) we reduced the number of rotations around z-axis in the training data from

This article has been accepted for publication in IEEE Transactions on Medical Imaging. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMI.2022.3218170

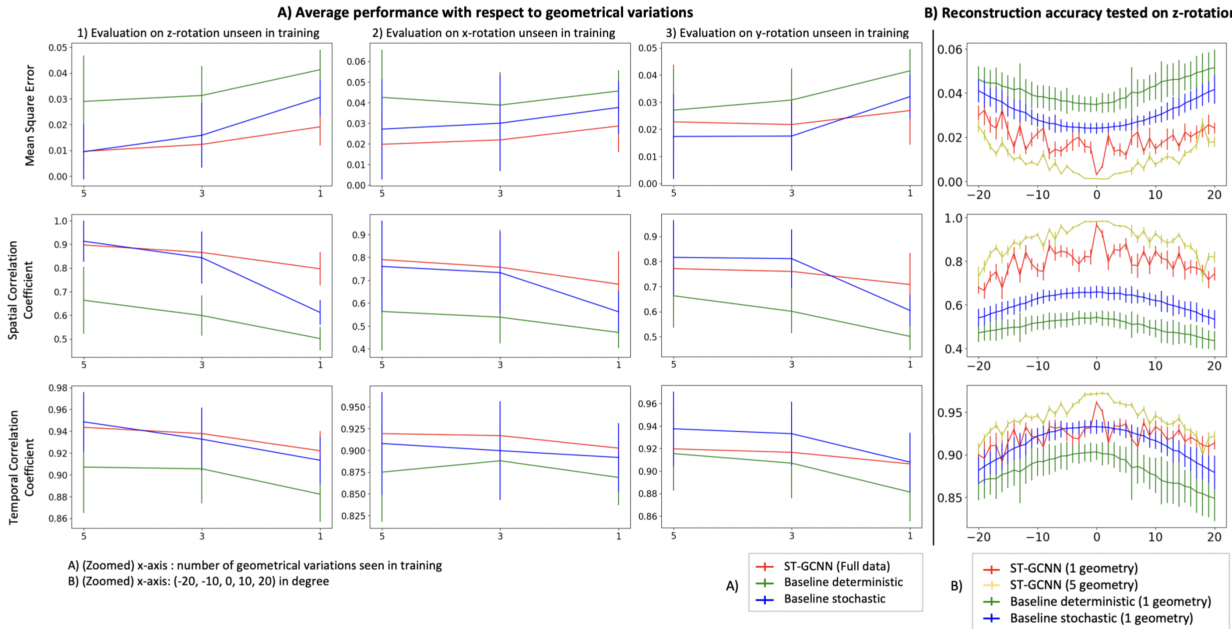

Fig. 5. Summary of average performance with respect to geometrical variations among the three comparison models.

-2° to 2° (five geometries), to -1° to 1° (three geometries), and eventually to no rotation at all (one single geometry).

1) Generalization with reduced diversity in training data: Fig. 3A1 shows quantitative metrics of the three models on z-axis rotations unseen in training set (21,771 cases), against the number of combinations of earliest activation sites and scar locations. The decrease in performance (reduced accuracy and increased standard deviation) in the face of reduced training data was significantly slower by our method (red) compared to the deterministic (green) and stochastic (blue) Euclidean baselines. Fig. 3A2-3 lists the performance of the three models on geometry with novel x- and y-rotations, showing a similar trend. While the performance of the ST-GCNN was similar and occasionally worse than the stochastic Euclidean baseline (*e.g.*, generalizing to y-rotations in Fig. 3A3) at the full diversity of the training data, it outperformed both baselines in all metrics as the training diversity decreased to 25%. In fact, its performance did not show notable change in performance until the number of training cases was reduced to below 5-10% of the complete set.

As an example, Fig. 3B shows the detailed quantitative metrics of the three models trained on 25% of the full training data, against x-axis rotations of the heart as measured by differences from training data. The presented method (red) outperformed the deterministic (green) and stochastic (blue) Euclidean baseline in all metrics, and its performance changed only slightly compared to that using full training data (yellow). Fig. 4 provides visual examples of the comparisons. The presented ST-GCNN model predicted the most accurate signal propagation pattern and scar location, while both baseline models showed errors on the free wall of the left ventricle.

2) Generalization with reducing training geometry: Fig. 5A1 summarizes quantitative metrics of the three models against the number of geometrical models in training. Our method (red) again showed a slower decrease in the average error and

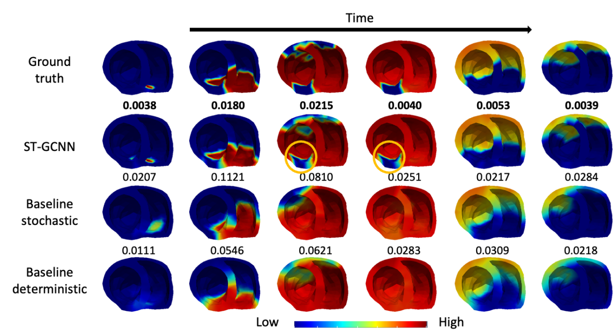

Fig. 6. Visual examples of reconstructed electrical activity trained on one single geometry and tested on the heart with a rotation at z = $\mathbf{10°}$. The MSE value is shown for each model at each timestep. Both baseline models showed substantial errors in locating the scar, while our model predicted the most accurate signal propagation pattern and the scar location (circles).

increase in standard deviation than the two Euclidean baselines. The testing results on novel x- and y-rotations showed a similar trend (Fig. 5A2-3). Note that these experiments were performed using training data that contained the full diversity, as described in the section above, thus the performance of ST-GCNN was lower than the stochastic Euclidean baseline when generalizing to y-rotations of the heart using five or three geometries.

Fig. 5B shows quantitative metrics of the three models when trained on a single geometry and tested on data following z-axis rotations of the heart. The decrease in performance compared to using more geometries (yellow curve) was more evident compared to the results in Fig. 3B, although it was much less significant compared to the two baselines. Fig. 6B provides visual examples of the reconstructed image sequences. Similar to the example shown in Fig. 4, ST-GCNN was the most accurate at predicting signal propagation and locating the scar, while both baseline models indicated incorrect scar locations.

C. Generalization to New Geometry

We then applied the model to 491 and 444 sets of simulated potentials generated on each of two new heart-torso meshes, respectively. This scenario is realistic in that the network trained on one group of patients will be applied to new patients. The blue flat line in Fig. 7 shows the accuracy of the results, trained on the full data set as described earlier and directly applied to the test data from the two new geometries. The performance (patient #2: MSE = 0.11, SCC = 0.23, TCC = 0.76; patient #3: MSE = 0.08, SCC = 0.19, TCC = 0.80) showed a larger decrease compared to those in Section IV-B, yet it is satisfactory overall. Note that Euclidean networks cannot be applied directly on a new geometry without re-training [14]. We then fine-tuned the model on a small set of data for each new geometry (122 and 111 sets, respectively), in comparison to retraining the model from scratch. As shown in Fig. 7, the fine-tuned model was much faster (< 100 epochs) to converge than the retrained model (> 300 epochs). Fig. 8 provides examples of the results. The fine-tuned model had the most accurate reconstruction of both the signal propagation pattern and the scar location among all the models we compared.

D. Training and Testing on Multiple Geometries

We further evaluated the ability of our model to be trained and tested across different geometries of the heart and torso, *i.e.*, across different input graphs. Such cross-training is not possible with Euclidean models unless the different heart and torso geometries are pre-registered. Even after registration, the learning will completely miss the different geometry information underlying the data. Therefore, in this set of experiments, we compared our method with traditional ECGI methods that rely on first building a forward mapping operator based on the given geometry, and then optimizing for the reconstruction, given the forward operator and applying second-order Tikhonov regularization [24]. Since unipolar extracellular potential, instead of transmembrane potential, is more commonly used as the source model in traditional ECGI methods, we chose to apply the presented method to reconstruct extracellular potential as well. Specifically, we considered the meshes in Section IV-A and Section IV-C for the simulation of extracellular potentials. We trained our model on a small subset of data for each geometry (50, 50, and 50 sets, respectively). We then evaluated the model on each geometry used in the training set but with rotations unseen in the training set. Fig. 10 shows that our method had a much better reconstruction accuracy than the ECGI method.

E. Effect of Stochastic IB Formulation

As observed in Section IV-B, in the Euclidean baselines, the stochastic model outperformed the deterministic one. This result was consistent with published results [14], which attributed this gain to the theory of information bottleneck in helping remove geometry-related information from the latent space. With our network trained as a function of the geometry, we expected that the stochastic IB formulation would bring less significant benefits. The results in Fig. 11B verified this assertion, as the performance of the stochastic model changed only minimally when using different values of β. Using $\beta = 1 \times 10^{-2}$ as an example (Fig. 11C), the performance gap between the stochastic and deterministic models was marginal.

TABLE I
COMPARISON OF ARCHITECTURES

Architecture Type	Time Complexity[a]	Average MSE[b]
L-Conv	12.1 min	0.0144
LSTM	29.5 min	0.0118
GCN-LSTM	32.6 min	0.0139
ST-GCNN	**9.5 min**	**0.0096**

[a] Average time per epoch.
[b] Average mean square error tested on all z-rotations with all models trained for 300 epochs. The other metrics followed the same trend.

F. Effect of Alternative Model Architectures

1) Temporal Modeling: We investigated several alternatives for temporal modeling in our network including: 1) interlaced spatial graph convolution and local temporal convolution [38] (L-Conv), which uses a 5×1 standard convolutional kernel to slide through the time sequence on each node and feature, 2) interlaced spatial graph convolution and regular Long Short-Term Memory (LSTM) networks [39] on temporal sequences, and 3) graph LSTM (GCN-LSTM), which replaces the fully connected operator in LSTM [39] with graph convolution so that the layer can operate on graph data. Table. I shows that the presented ST-GCNN model was the most efficient and achieved the best performance.

2) Residual Blocks: We further trained a geometric network without residual blocks on the dataset used in Section IV-A and tested it on the same dataset of rotations as described in Section IV-B. Fig. 11A summarizes the mean square error of two networks against the change in heart rotations from the training data. As shown, without the residual blocks, the network was not able to accurately reconstruct heart potentials.

3) Latent inverse mapping: We investigated a more general modeling option at the latent space for the relationship between latent embedding $\mathbf{Z}_h$ and $\mathbf{Z}_b$: a fully-connected layer with ELU activation [41]. This describes $\mathbf{Z}_h$ as a general nonlinear function of $\mathbf{Z}_b$ without considering the underlying geometry. We trained this network on 50 data samples on one single geometry and tested it on 3,095 data samples from the same geometry but with five different rotations of the heart. As shown in Fig. 12, the resulting network – while performing well on the training geometry – struggled with testing data from different geometries.

To further understand the learned latent inverse mapping in (4), we examined empirical evidence on how the coefficient of the linear inverse mapping $\hat{\mathbf{h}}(\mathbf{u}(i,j))$ changed with the relative distance between vertex i and j in the latent torso and heart graphs. Due to the high dimensionality of $\mathbf{z}_h$ and $\mathbf{z}_b$, this was difficult to observe directly. Instead, we set all $\mathbf{z}_b(j)$ on the torso graphs to be identical constants, chose random vertex i on the heart graph, and obtained the norm of $\hat{\mathbf{h}}(\mathbf{u}(i,j))$

This article has been accepted for publication in IEEE Transactions on Medical Imaging. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMI.2022.3218170

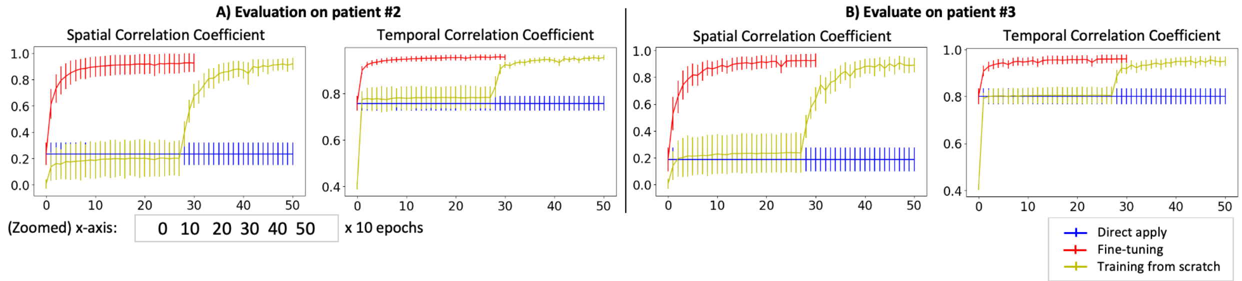

Fig. 7. Convergence of reconstruction accuracy on new geometry by the fine-tuned (red) *vs.* retrained model (yellow).

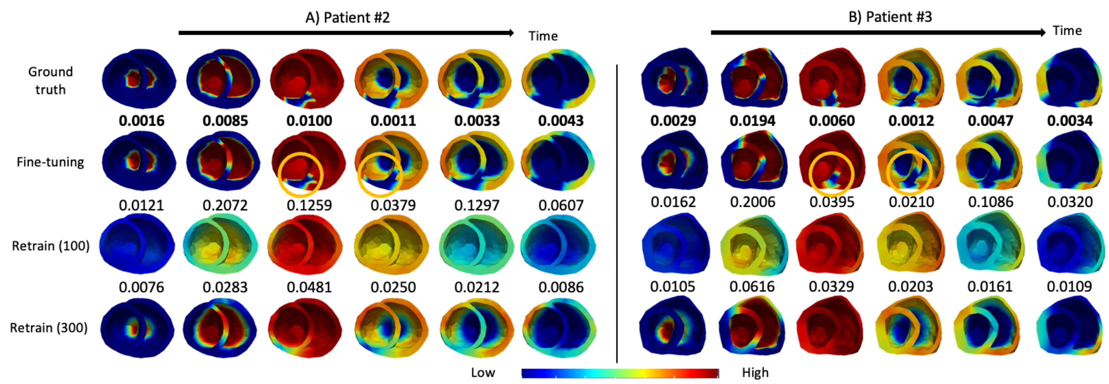

Fig. 8. Examples of reconstructed electrical activity by the fine-tuned model at epoch = 100, the retrained model at epoch = 100, and the retrained model at epoch = 300 on A) patient #2 and B) patient #3. The MSE value is shown for each model at each timestep. The fine-tuned model has the most accurate reconstruction of the signal propagation pattern and the scar location (circles).

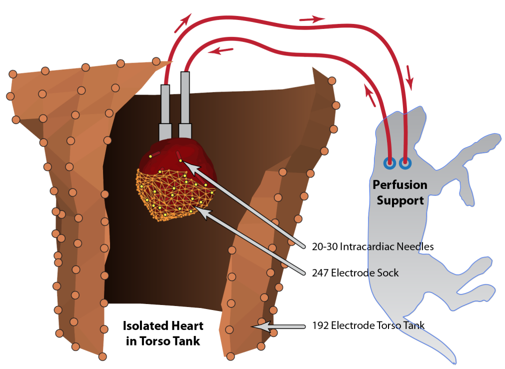

Fig. 9. A flexible epicardial sock array encircled the isolated heart, which was perfused from a second support animal through the aorta with blood returned under suction from the right ventricle. The human-torso-shaped tank was filled with electrolytic fluid consistent with human torso conductivity and contained 192 embedded Ag/AgCl electrodes. The recording system sampled cage and torso potentials simultaneously. Bipolar stimulation was initiated from intramural plunge needles

for each j which describes how each $\mathbf{z}_b(j)$ contributed to $\mathbf{z}_h(i)$ for different vertices j. Fig. 13 shows three examples of randomly selected i on the heart graph: consistent with the known physics, the contribution of each $\mathbf{z}_b(j)$ to $\mathbf{z}_h(i)$ decreased as the distance of (i, j) increased. This suggests that the inverse mapping learned geometry-dependent functions consistent with the underlying physics.

V. Real Data Experiments

A. Experimental Data Description

1) Torso Tank Experimental Preparation: The experimental data sets used in this study were acquired from a modified Langendorff-perfused torso tank preparation [44]. As illustrated in Fig. 9, an isolated canine heart was suspended within a human shaped torso tank and perfused via arterial blood from a second support dog. Blood was returned to the support dog from a right ventricular cannula to the jugular vein. The human-shaped torso tank was filled with an electrolytic solution (resistivity was 500 Ω-cm), which approximates the electrical conductivity of a human torso. The animals were under deep anesthesia using procedures approved by the Institutional Animal Care and Use Committee of the University of Utah and conformed to the Guide for the Care and Use of Laboratory Animals.

2) Signal Acquisition: Cardiac activation was generated with bipolar stimulation using plunge needles at five sites: left ventricular (LV) base, LV Apex, LV freewall, LV septum, and right ventricular (RV) free wall. All stimulation was initiated near the endocardium and signals were recorded for five seconds.

Cardiac potentials were recorded using an epicardial sock with 247 electrodes (inter-electrode spacing 6.5$\pm$1.3 mm) stretched over the ventricles of the heart. The torso tank had 192 silver/silver-chloride electrodes (with inter-electrode spacing 40.2$\pm$16.8 mm) distributed across the outer surface. All signals were referenced to a Wilson's Central Terminal and were simultaneously sampled at 1000 Hz. Signals were filtered, annotated, and post-processed using PFEIFER [45].

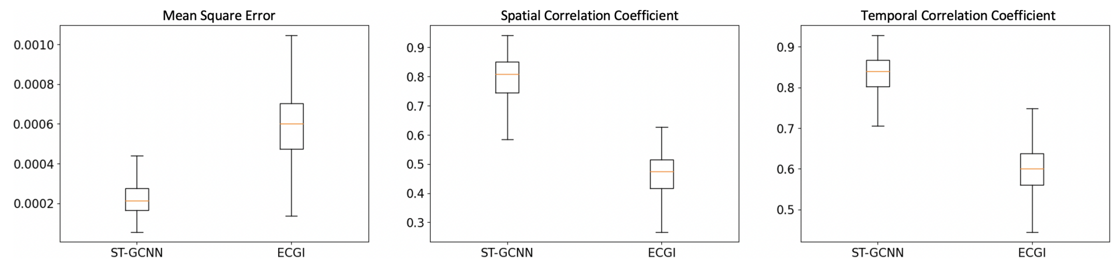

Fig. 10. Performance of our ST-GCNN method *vs.* ECGI.

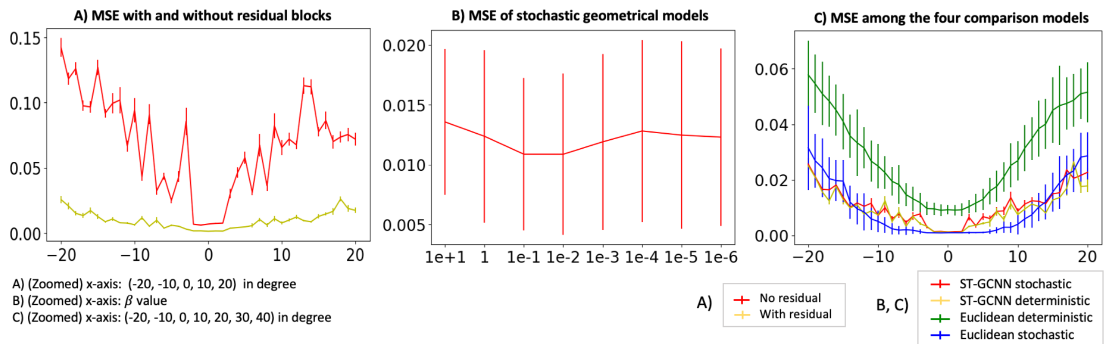

Fig. 11. A) Effect of residual blocks; B) Effect of hyperparameter value of β; C) Comparison of model performance at $\beta = 0.01$.

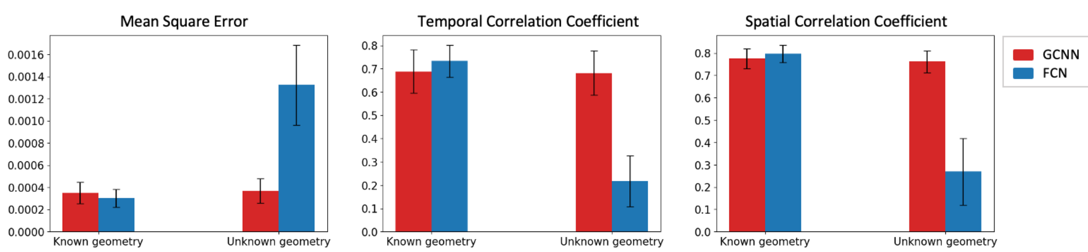

Fig. 12. Performance of GCNN and non-linear fully-connected (FCN) latent inverse mapping.

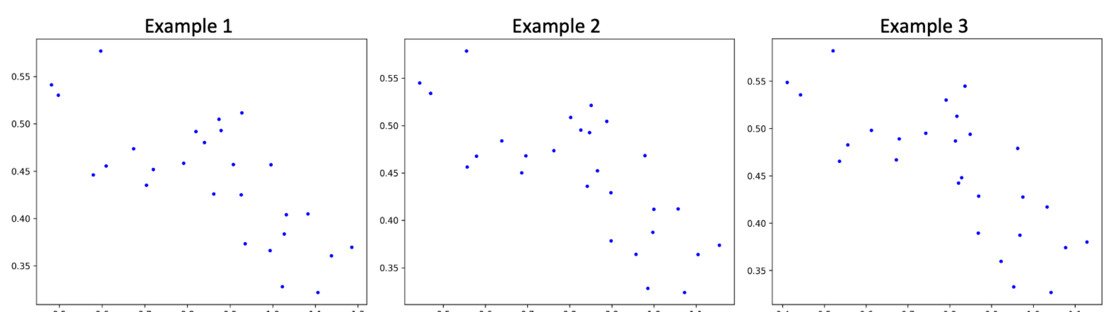

Fig. 13. Examples of the relation between the norm of $\mathbf{z}_b(j) \cdot \hat{\mathbf{h}}(\mathbf{u}(i,j))$ (y-axis) and the distance $\mathbf{u}(i,j)$ between vertex i and j on the bipartite graph (x-axis).

3) Geometric Model Creation: The surface geometries were constructed based on electrode locations acquired during each experiment. Template geometries for both the torso tank and epicardial sock were registered using known correspondence points, which were measured using a 3D mechanical digitizer (Microscribe from Immersion Corp). The epicardial sock registration was further refined as described previously [46].

B. Evaluation & Results

120 out of 192 torso-tank measurements were selected for inverse imaging, to be consistent with the number of input measurements used in the synthetic training data. The measured QRST signals were downsampled using polyphase filtering to the length of the simulated training signals. The epi-endocardial geometry used in Section IV-A was registered to the epicardial sock geometry with transition, rotation, and scaling operations. The measured epicardial potential and the sites of stimulation were registered to this epi-endocardial model, to provide reference data for evaluation. The inverse imaging results were evaluated by MSE, SCC, and TCC metrics against the measured epicardial potentials.

We carried out cross validation by leaving out the signals

This article has been accepted for publication in IEEE Transactions on Medical Imaging. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/TMI.2022.3218170

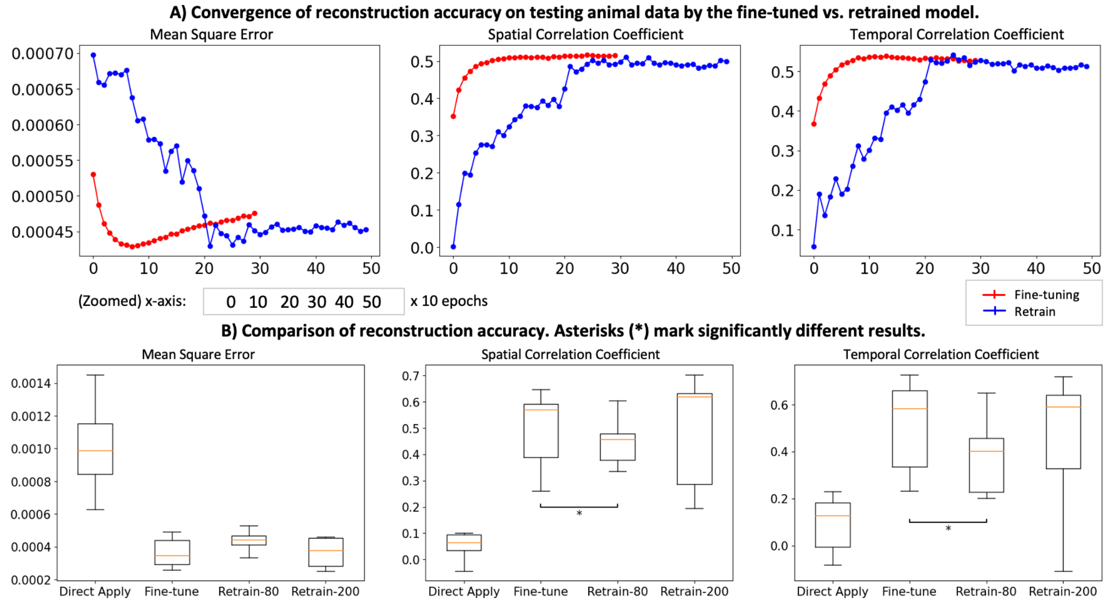

Fig. 14. Performance of comparison models on measured data: A) Convergence of fine-tuned *vs.* retrained model; B) Final accuracy of various models.

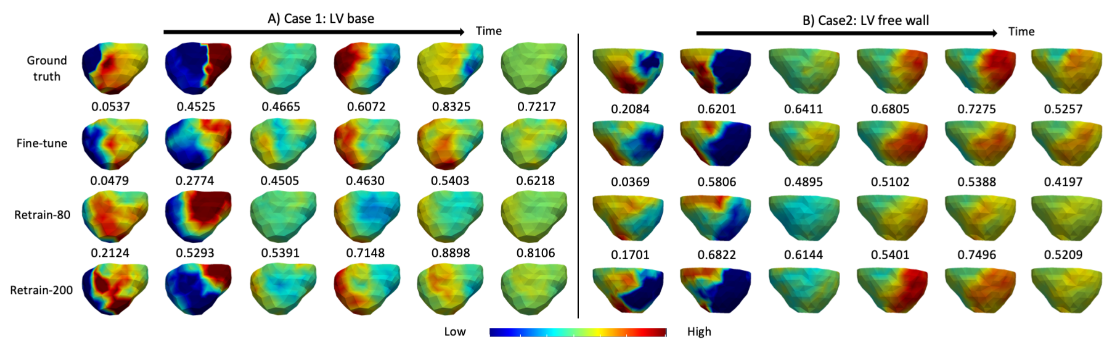

Fig. 15. Examples of reconstructed electrical activation sequences stimulated from A) LV base and B) LV free wall. The SCC value is shown for each model at each timestep. The fine-tuned model and the retrained model at epoch = 200 had similar performance.

from one stimulated activation sequence (5 sequences) each time. We first directly applied a model trained on simulated data, considering a subset from those described in Section IV-A including 72 different combinations of activation origins and scar tissues. We then fine-tuned the trained model using measured signals from the remaining four stimulation sites (20 sequences). Finally, we retrained the same model from scratch using the same experimental data. Fig. 14A illustrates the quantitative metrics on the test data, averaged over the cross-validation folds as the training of the fine-tuned and re-trained models converged. Similar to the results presented in Section IV-C, the fine-tuned model took many fewer epochs (80 epochs) to converge than the retrained model (200 epochs). Fig. 14B summarizes quantitative metrics obtained by these models. As shown, all metrics of the fine-tuned model were significant better than the retrained model at epoch 80 ($p = 0.05$ (MSE), 0.013 (SCC), and 0.001 (TCC), paired t-tests), and moderately better than the retrained model after convergence ($p = 0.32$ (MSE), 0.10 (SCC), and 0.14 (TCC), paired t-tests). Fig. 15 provides visual examples of two paced activations. The fine-tuned model and the retrained model at epoch = 200 had similar performance on the prediction of propagation of activation.

VI. Discussion and Conclusions

We have presented a novel non-Euclidean network for learning geometry-dependent and physics-based inverse mapping between spatiotemporal variables mapped to 3D geometrical domains. We demonstrated its ability to improve generalization to unseen geometrical variations in comparison to its Euclidean alternatives, to directly apply to new geometry in a way that is not possible with Euclidean approaches, and to be quickly fine-tuned to a new geometry using a small amount of data. To our knowledge, this is the first report of a geometry-dependent non-Euclidean inverse imaging network. Our method is general for problems with spatiotemporal data living on graphs and linked with a linear imaging operator. Future studies will extend its application to other problems that fall into this category as well as to incorporate more general physics. Furthermore, as observed in the results using data from experiments, there is still a performance gap between models based on simulation and measured data, even after fine-tuning. Given the challenges in obtaining labeled data in the types of application considered in this study (*e.g.*, measurement of whole-heart electrical potential), there is considerable motivation to investigate how to supervise the network with the governing physics in addition to data-driven losses.

Several aspects of the presented ST-GCNN could also be further investigated. One of the challenging factors lies in the hierarchical graph representations of the heart-torso mesh. One critical consideration in this problem is to prevent non-physical spatial propagation of signals, which requires the coarsened graph in hierarchical graph representations to preserve the topology of the geometry. Unlike down-sampling strategies on 2D Euclidean spaces, there is no established automatic method for down-sampling node features on realistic geometrical spaces. Existing graph pooling methods use clustering method based on graph topological structure to coarsen the graph [47], [48]. However, we found significant structural information loss that, for instance, introduced holes on the right ventricle (RV) of the coarsened heart mesh. We also noticed that the activation did not always propagate over the surface on the coarsened heart. Therefore, we adopted a specialized mesh-coarsening approach from CGAL [37], to preserve the topological information of the coarsened heart and torso mesh. This mesh coarsening method also allowed us to control the down-sampling rate to prevent the unacceptable loss of structural information.

There are various choices of source models to represent the electrical activity of the heart in existing ECGI approaches, including heart surface potentials [11]–[13], or transmembrane potential defined on the volumetric mesh of the heart [6], [7], [14], [19], [49]. We based this study on the former because this is the most common formulation and the one used in commercial systems and also because surface-based methods are more straightforward to implement. Any further extension of the ST-GCNN to the more complete volumetric representation of the cardiac electrical sources will require appropriate hierarchical graph representations, which will certainly become more challenging. Furthermore, the size of the graph and thus the computational cost of training the ST-GCNN can also be expected to increase substantially.

There are many sources of geometric variation that we did not evaluate; in addition to variations between subjects, breathing can expand and contract the torso and alter the location of the heart relative to the torso; the heart geometry and position changes every time the heart beats; regular human posture and activities may also cause slight transition or rotation of the heart. Considering all these scenarios of geometrical variation in any data-driven method could be challenging, given the volume and variation in the required geometric and signal information required for training. In this study, we simplified the geometrical variation to heart rotations, following the most common settings in the previous reports [14].

Acknowledgements

This study was supported by NIH NHLBI grant R01HL145590 (Wang), NSF CAREER Award ACI-1350374 (Wang), NIH NIGMS Center for Integrative Biomedical Computing (www.sci.utah.edu/cibc), NIH NIGMS grants P41GM103545 and R24GM136986 (MacLeod), and the Nora Eccles Treadwell Foundation for Cardiovascular Research (MacLeod).

References

[1] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," IEEE transactions on image processing, vol. 26, no. 7, pp. 3142–3155, 2017.
[2] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: beyond analytical methods," IEEE Signal Processing Magazine, vol. 35, no. 1, pp. 20–36, 2018.
[3] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 2, pp. 295–307, 2015.
[4] B. Zhu, J. Z. Liu, S. F. Cauley, B. R. Rosen, and M. S. Rosen, "Image reconstruction by domain-transform manifold learning," Nature, vol. 555, no. 7697, p. 487, 2018.
[5] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep admm-net for compressive sensing mri," in Proceedings of the 30th international conference on neural information processing systems, pp. 10–18, 2016.
[6] S. Ghimire, J. Dhamala, P. K. Gyawali, J. L. Sapp, M. Horacek, and L. Wang, "Generative modeling and inverse imaging of cardiac transmembrane potential," in International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 508–516, Springer, 2018.
[7] S. Ghimire, J. L. Sapp, B. M. Horáček, and L. Wang, "Noninvasive reconstruction of transmural transmembrane potential with simultaneous estimation of prior model error," IEEE transactions on medical imaging, vol. 38, no. 11, pp. 2582–2595, 2019.
[8] C. M. Michel and M. M. Murray, "Towards the utilization of eeg as a brain imaging tool," Neuroimage, vol. 61, no. 2, pp. 371–385, 2012.
[9] T. Bacoyannis, J. Krebs, N. Cedilnik, H. Cochet, and M. Sermesant, "Deep learning formulation of ecgi for data-driven integration of spatiotemporal correlations and imaging information," in International Conference on Functional Imaging and Modeling of the Heart, pp. 20–28, Springer, 2019.
[10] M. Jiang, F. Liu, Y. Wang, G. Shou, W. Huang, and H. Zhang, "A hybrid model of maximum margin clustering method and support vector regression for noninvasive electrocardiographic imaging," Computational and Mathematical Methods in Medicine, vol. 2012, 2012.
[11] A. Malik, T. Peng, and M. L. Trew, "A machine learning approach to reconstruction of heart surface potentials from body surface potentials," in 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 4828–4831, IEEE, 2018.
[12] A. Karoui, M. Bendahmane, and N. Zemzemi, "A spatial adaptation of the time delay neural network for solving ecgi inverse problem," in International Conference on Functional Imaging and Modeling of the Heart, pp. 94–102, Springer, 2019.
[13] J. Horvath, L. Shien, T. Peng, A. Malik, M. Trew, and L. Bear, "Deep learning neural nets for detecting heart activity," arXiv preprint arXiv:1901.09831, 2019.
[14] S. Ghimire, P. K. Gyawali, J. Dhamala, J. L. Sapp, M. Horacek, and L. Wang, "Improving generalization of deep networks for inverse reconstruction of image sequences," in International Conference on Information Processing in Medical Imaging, pp. 153–166, Springer, 2019.
[15] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond euclidean data," IEEE Signal Processing Magazine, vol. 34, no. 4, pp. 18–42, 2017.
[16] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," IEEE Transactions on Neural Networks and Learning Systems, 2020.
[17] X. Jiang, S. Ghimire, J. Dhamala, Z. Li, P. K. Gyawali, and L. Wang, "Learning geometry-dependent and physics-based inverse image reconstruction," in International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 487–496, Springer, 2020.
[18] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," arXiv preprint physics/0004057, 2000.
[19] L. Wang, H. Zhang, K. C. Wong, H. Liu, and P. Shi, "Physiological-model-constrained noninvasive reconstruction of volumetric myocardial transmembrane potentials," IEEE Transactions on Biomedical Engineering, vol. 57, no. 2, pp. 296–315, 2009.

[20] C. Ramanathan, R. N. Ghanem, P. Jia, K. Ryu, and Y. Rudy, "Noninvasive electrocardiographic imaging for cardiac electrophysiology and arrhythmia," Nature medicine, vol. 10, no. 4, pp. 422–428, 2004.
[21] R. Plonsey, "Bioelectric phenomena," Wiley Encyclopedia of Electrical and Electronics Engineering, 2001.
[22] R. Barr, M. Ramsey, and M. Spach, "Relating epicardial to body surface potential distributions by means of transfer coefficients based on geometry measurements," IEEE Trans. Biomed. Eng., vol. 24, pp. 1–11, Jan. 1977.
[23] A. Pullan, L.K.Cheng, M. Nash, D. Brooks, A. Ghodrati, and R. MacLeod, "The inverse problem of electrocardiography," in Comprehensive Electrocardiology (P. Macfarlane, A. van Oosterom, O. Pahlm, P. Kligfield, M. Janse, and J. Camm, eds.), pp. 299–344, London, UK: Springer Verlag, 2010.
[24] Y. Rudy and H. Oster, "The electrocardiographic inverse problem.," Critical reviews in biomedical engineering, vol. 20, no. 1-2, p. 25, 1992.
[25] D. H. Brooks, G. F. Ahmad, R. S. MacLeod, and G. M. Maratos, "Inverse electrocardiography by simultaneous imposition of multiple constraints," IEEE Transactions on Biomedical Engineering, vol. 46, no. 1, pp. 3–18, 1999.
[26] S. Ghosh and Y. Rudy, "Application of l1-norm regularization to epicardial potential solution of the inverse electrocardiography problem," Annals of biomedical engineering, vol. 37, no. 5, pp. 902–912, 2009.
[27] J. Xu, A. R. Dehaghani, F. Gao, and L. Wang, "Noninvasive transmural electrophysiological imaging based on minimization of total-variation functional," IEEE transactions on medical imaging, vol. 33, no. 9, pp. 1860–1874, 2014.
[28] P. M. Van Dam, T. F. Oostendorp, A. C. Linnenbank, and A. Van Oosterom, "Non-invasive imaging of cardiac activation and recovery," Annals of biomedical engineering, vol. 37, no. 9, pp. 1739–1756, 2009.
[29] R. Hoekema, G. J. Uijen, and A. Van Oosterom, "Geometrical aspects of the interindividual variability of multilead ecg recordings," IEEE Transactions on Biomedical Engineering, vol. 48, no. 5, pp. 551–559, 2001.
[30] S. Giffard-Roisin, H. Delingette, T. Jackson, J. Webb, L. Fovargue, J. Lee, C. A. Rinaldi, R. Razavi, N. Ayache, and M. Sermesant, "Transfer learning from simulations on a reference anatomy for ecgi in personalized cardiac resynchronization therapy," IEEE Transactions on Biomedical Engineering, vol. 66, no. 2, pp. 343–353, 2018.
[31] M. Fey, J. Eric Lenssen, F. Weichert, and H. Müller, "Splinecnn: Fast geometric deep learning with continuous b-spline kernels," in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 869–877, 2018.
[32] Y. Seo, M. Defferrard, P. Vandergheynst, and X. Bresson, "Structured sequence modeling with graph convolutional recurrent networks," in International Conference on Neural Information Processing, pp. 362–373, Springer, 2018.
[33] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," arXiv preprint arXiv:1709.04875, 2017.
[34] S. Cao, W. Lu, and Q. Xu, "Deep neural networks for learning graph representations," in Thirtieth AAAI conference on artificial intelligence, 2016.
[35] T. N. Kipf and M. Welling, "Variational graph auto-encoders," arXiv preprint arXiv:1611.07308, 2016.
[36] M. Simonovsky and N. Komodakis, "Graphvae: Towards generation of small graphs using variational autoencoders," in International Conference on Artificial Neural Networks, pp. 412–422, Springer, 2018.
[37] F. Cacciola, "Triangulated surface mesh simplification," in CGAL User and Reference Manual (C. E. Board, ed.), 3.3 ed., 2007.
[38] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[40] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114, 2013.
[41] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," arXiv preprint arXiv:1511.07289, 2015.
[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.
[43] R. R. Aliev and A. V. Panfilov, "A simple two-variable model of cardiac excitation," Chaos, Solitons & Fractals, vol. 7, no. 3, pp. 293–301, 1996.
[44] J. Bergquist, J. Coll-Font, B. Zenger, L. Rupp, W. Good, D. Brooks, and R. MacLeod, "Improving localization of cardiac geometry using ecgi.," in Comput in Cardiol 2020, vol. 47, pp. 1–4, Sep 2020.
[45] A. Rodenhauser, W. Good, B. Zenger, J. Tate, K. Aras, B. Burton, and R. MacLeod, "PFEIFER: Preprocessing framework for electrograms intermittently fiducialized from experimental recordings," J. Open Source Software, vol. 3, p. 472, Sept. 2018.
[46] J. Coll-Font and D. H. Brooks, "Tracking the position of the heart from body surface potential maps and electrograms," Frontiers in Physiology, vol. 9, p. 1727, 2018.
[47] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors a multilevel approach," IEEE transactions on pattern analysis and machine intelligence, vol. 29, no. 11, pp. 1944–1957, 2007.
[48] M. Simonovsky and N. Komodakis, "Dynamic edge-conditioned filters in convolutional neural networks on graphs," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3693–3702, 2017.
[49] J. Dhamala, S. Ghimire, J. L. Sapp, B. M. Horáček, and L. Wang, "Bayesian optimization on large graphs via a graph convolutional generative model: Application in cardiac model personalization," in International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 458–467, Springer, 2019.