Left Ventricle Motion Estimation in Cine MRI With Multilayer Iterative Deformable Graph Matching

ZHENGRUI ZHANG[ID]**1, XUAN YANG2, JUNHAO WU2, AND GUOLIANG CHEN2**
1College of Information Engineering, Shenzhen University, Shenzhen 518000, China
2College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518000, China

Corresponding author: Xuan Yang (yangxuan@szu.edu.cn)

ABSTRACT Quantifying regional myocardial motion and deformation from cardiac magnetic resonance imaging (MRI) plays an important role in clinical applications. In this paper, we present a novel approach for the estimation of left ventricle myocardial motion based on iterative deformable graph matching for cine MRI. Graph matching is a shape matching approach that can be used to determine the correspondence between two objects. However, existing graph matching algorithms are unsuitable for applications with large deformations. In this paper, we propose an iterative deformable graph matching framework for estimating the correspondence between points extracted from left ventricle myocardium at the end-diastolic and end-systolic phases to estimate cardiac motion. A new cost function for graph matching is defined to measure the discrepancy between the nodes and edges of two graphs under a transformation. By introducing a spatial transformation with a sparsity constraint, we can estimate a robust deformation field, alleviating the influence of inevitable graph mismatches. The correspondence between points is then updated by mapping the source graph using the estimated transformation. The cost function is optimized by alternatively optimizing for correspondence and spatial transformation. Furthermore, we propose a multilayer framework to improve correspondence accuracy using a bottom-up matching procedure. This framework estimates the deformation field between an image at the end-systolic phase and an image at the end-diastolic phase in an MRI sequence. Evaluations of two public cardiac datasets indicate that the proposed framework outperforms traditional graph matching algorithms in accuracy and robustness.

INDEX TERMS Motion estimation, graph matching, correspondence, deformation field.

I. INTRODUCTION

Cardiovascular diseases (CVDs) are the leading cause of death in the world, and it is predicted that the annual number of deaths due to CVDs will increase from 17 million in 2008 to 25 million in 2030 [1]. Heart diseases such as cardiomyopathy and ischemia can affect the structure and function of the left ventricle (LV) over the cardiac cycle. Understanding cardiac anatomy and function is crucial in order to diagnose and treat CVDs. Moreover, analyzing the LV structure with imaging instruments is effective at reducing mortality and morbidity associated with CVDs. Quantitative assessments of ventricular wall motion, such as its volume output [2] and strain [3], facilitate better understanding of

The associate editor coordinating the review of this manuscript and approving it for publication was Vincenzo Piuri.

CVDs and their treatment. Estimations of cardiac motion constitute an important aid when quantifying the elasticity and contractility of the myocardium.

Analyzing LV motion relies on noninvasive imaging techniques. Magnetic resonance imaging (MRI) is one such technique that supplies accurate information on cardiac morphology, muscle perfusion, tissue viability, and blood flow, such that global and regional changes to the LV can be observed. MRI allows for an analysis of LV structural alterations and can be used to measure functional changes to the LV [4]. Advanced MRI equipment can obtain cardiac displacement fields and derived parameters such as myocardial strain with high accuracy [5]. Tagged MRI uses noninvasive markers in the myocardial tissue during the cardiac cycle, superimposing a grid of tag lines on the myocardium region that can be tracked to estimate the motion of the

LV myocardium [6], [7]. However, tagged MRI is not a conventional imaging protocol for LV function assessment, and the tags fade during the cardiac cycle, making it difficult to identify tags in the last phase of the sequence of images [8], [9]. Compared to tagged MRI, cine MRI is widely used, owing to its high soft-tissue contrast and temporal resolution. It has the advantage of revealing the cardiac anatomy relatively clearly, especially the myocardium. Unfortunately, MRI cannot directly provide information regarding myocardium motion as the myocardium appears highly homogeneous and there are few features inside it.

In this paper, we focus on LV motion estimation for the clinically useful cine MRI technique. The various LV motion estimation methods that have previously been proposed for cine MRI can be divided into image registration and feature tracking methods. The former methods optimize a cost function between two images taken at different time points, then employ a spatial transformation function to compute the displacement of the myocardium [6], [10]–[16]. Feature tracking methods generally track selected features of the myocardium wall [17]–[22]. The most commonly used features in cine MRI are the endocardial and epicardial contours or specific feature points on these contours [23]. Additional constraints are needed with feature tracking methods when estimating a dense motion field from a sparse set of feature points: e.g., incompressibility [24], [25], and smoothness and periodicity constraints [26]. In feature tracking methods, point set matching [27]–[30] is a commonly used algorithm that can estimate the correspondence and transformation function between two-point sets simultaneously. However, point set matching exclusively considers the distance between point sets, not the topological relationship between points, which can easily lead to abnormal correspondence between points extracted from the myocardium. Graph matching (GM) is a feature matching method that represents objects using graphs and finds a correspondence between the nodes of two graphs. GM estimates not only the point-to-point correspondence but also the edge-to-edge consistency between two graphs. It plays an important role in solving correspondence problems in computer vision.

Point sets are commonly used to represent objects. In this case, because the point sets have important internal structures, they are often considered not simply as point sets, but as separate graphs. In this setting, graph nodes represent feature points extracted from images and graph edges represent relationships between feature points [31]. The goal of GM is to find a correspondence between the nodes of two graphs such that they are similar to each other when their vertices are labeled according to such a correspondence. GM algorithms are optimization techniques that consider these two types of information to compute the matches, instead of only relying on a simple nearest neighborhood relationship, as in point set matching. For LV motion estimation for short-axis MRIs, the myocardium appears highly homogeneous and there are few features inside the myocardium in cine MRI, which makes it difficult to determine the correspondence between the contours of the myocardium by relying solely on point features. One reasonable choice for improving the accuracy of correspondence estimation is to consider the relationships between points. As mentioned above, GM algorithms estimate point correspondence not only based on the point features but also the edge-to-edge consistency between two graphs, which makes it possible to improve the correspondence accuracy between points located on contours of the LV myocardium. Consequently, we employ GM algorithms to estimate the correspondence between points extracted from the LV myocardium and estimate the dense cardiac motion fields using interpolation functions.

In GM, estimating the correspondence between two graphs is essentially an NP-hard quadratic assignment problem. Research on GM focuses on finding approximate solutions that relax the constraints of the corresponding matrix to find an approximate solution. Based on the approximate strategy adopted, GM can be roughly classified into three categories [32]: spectral relaxation, semi-definite programming, and doubly stochastic matrix relaxation.

Spectral relaxation treats the GM problem as a Koopmans–Beckmann quadratic assignment problem, by looking for a permutation matrix to render two graphs isomorphic. Spectral relaxation derives from the spectral method, which decomposes the adjacency matrix of the graph and selects eigenvectors corresponding to large eigenvalues as solutions. Scott and Longuet-Higgins [33] used the spatial relationship between the vertices in a graph to define a neighboring matrix and solve the permutation matrix via eigenvalue decomposition. Shapiro and Brady [34] proposed a method that overcomes the rotation sensitivity in [33], although their method requires the same number of vertices in both graphs. Umeyama [35] assumed that the corresponding matrix is orthogonal and approximated the permutation matrix as an orthogonal matrix. Leordeanu and Hebert [36] relaxed the orthogonality of the permutation matrix and restricted the values of the permutation matrix to between 0 and 1. Their method involves using a greedy algorithm to select the eigenvectors corresponding to larger eigenvalues, making it more robust to outliers and noise. Based on the method by Leordeanu *et al.*, Cour [31] introduced the affine constraint to spectral relaxation, which requires that the sum of each row and each column of the corresponding matrix is one. Compared to the spectral method, Cour's method more closely approximates the original quadratic program problem. Leordeanu *et al.* [37] proposed an iterative matching method that optimizes the quadratic assignment problem in a discrete domain and strictly satisfies one-to-one mapping constraints during optimization. Leng *et al.* [38] defined a normalized stochastic Laplacian matrix for weighted graphs that can be solved using principal components decomposition.

Semi-definite programming (SDP) is a type of convex optimization that maximizes a linear function while satisfying the affine constraint. Schellewald and Schnörr [39] regarded GM as a bipartite GM problem and constructed an SDP cost

function with positive integer solutions. Bai *et al.* [40] transformed GM into point matching in metric space and used the SDP to solve the GM model proposed by Scott and Longuet-Higgins [33].

Doubly stochastic matrix relaxation methods relax the corresponding matrix such that it becomes a doubly stochastic matrix, with nonnegative entries and with row sums and column sums equal to one, guaranteeing a one-to-one correspondence. A double stochastic matrix is a closed convex set, and when a linear function is minimized over doubly stochastic matrices, a solution can always be found in the set of permutation matrices [41]. The solutions can be obtained at the vertex of the double stochastic matrix, which is a permutation matrix. The double stochastic matrix constraint makes GM a non-convex quadratic programming problem, which can be approximated using the path-following algorithm [32], [41], [42]. The path following algorithm, first proposed by Zaslavskiy *et al.* [41], discretizes the values of a doubly stochastic matrix. Liu *et al.* [42] used the path following algorithm to solve directed GM. Zhou and De la Torre [32] decomposed the affinity matrix of GM and employed the path-following algorithm to solve the corresponding matrix. Jiang *et al.* [43] formulated GM as an optimization problem with nonnegative and sparse constraints, employing a nonnegative matrix factorization algorithm to solve the constrained problem. These sparsity constraints are similar to the doubly stochastic matrices and can lead to a sparse solution.

GM is robust at structural matching, which is especially useful when matching two similar objects. To date, most researchers have been committed to improving correspondence precision by adopting various optimization algorithms and constraints. However, in many applications, an affine transformation between two-point sets is assumed for GM algorithms. Matching problems in computer vision naturally require global constraints among nodes in the graph, including affine and elastic deformations. However, most existing GM algorithms can only cope with matching problems with affine deformation constraints. When matching two objects with large elastic deformations, the corresponding edges are considerably distorted and the similarity between edges is low, which results in most GM algorithms being unsuitable for applications with large deformations. For LV motion estimation, a large elastic deformation exists between the LV at the end-systolic (ES) and end-diastolic (ED) phases. Consequently, GM algorithms cannot accurately estimate the correspondence between two-point sets extracted from the boundaries of the LV at the ES and ED phases, respectively.

Most existing GM algorithms do not constrain the graphs to be mapped with a spatial transformation; hence, matching accuracy is affected in cases with large deformation. Berg *et al.* [44] combined the similarity of point descriptors and geometric distortion in a cost function defined over the correspondence space, finding a trade-off between rotation and scale invariance in shape matching. Furthermore, deformable GM has been proposed [45] to constrain the nodes of both graphs to a given geometric transformation. However, only traditional rigid or nonrigid transformations, such as a thin plate spline (TPS), are used in [45], and these transformations are sensitive to mismatched nodes. As such, inevitable graph mismatches will result in unexpected spatial transformations. These, in turn, will cause GM errors during subsequent iterations.

Solving the correspondence problem between objects and estimating the spatial transformation between objects are important tasks for LV motion estimation. The matching problems in LV motion estimation naturally require spatial transformation constraints between LVs at different times. When applying LV motion estimation, the myocardium appears highly homogeneous; further, there are few features inside the myocardium in cine MRI, making it difficult to determine the correspondence between contours of the myocardium. The central aim of LV motion estimation is to estimate the correspondence between sampled points on the LV contours extracted from images at different times. Because the deformation between images at the ES and ED phases in an MRI sequence is considerable, existing GM algorithms cannot be employed directly to estimate anatomical correspondence.

In this paper, we present a new method for LV motion estimation in which a GM algorithm is alternated with robust transformation estimation. Corresponding points located on the LV contours can be provided by GM. To cope with mismatched point pairs in GM, a robust transformation model is employed. The source graph is then mapped by the transformation and a new correspondence between the points can be estimated iteratively. The contributions of this study are as follows:

(1) A robust transformation model [46] is introduced to the GM algorithm. This transformation model is constrained by l_1- and l_2-norm regularizations. The model is robust to landmark location errors and can be used to estimate a stable deformation field, which is especially useful when estimating the deformation field between two LV objects when mismatched point pairs exist in GM.

(2) We present a multilayer framework that performs GM and transformation estimation alternately in a bottom-up manner. We sample the graph nodes sparsely and estimate the transformation on the sampled graph. After mapping the source graph using the estimated transformation, dense graph nodes are sampled and a new transformation is estimated. By taking advantage of this multilayer framework, we can obtain highly accurate matching results with this bottom-up procedure.

(3) Our method is applied to LV motion estimation for cine MRI. Evaluations on two public datasets show that our method considerably improves GM accuracy and obtains more accurate LV motion fields.

We introduce the GM problem briefly in Section II. A new method that combines GM with a robust transformation model is presented in Section III and our proposed multilayer framework is described in Section IV. The computational

complexity of the proposed framework is analyzed in Section V. In Section VI, we outline the evaluation conducted of our proposed framework on public cine MR cardiac images. Finally, conclusions are presented in Section VII.

II. GRAPH MATCHING

In this section, we offer a brief overview of the GM problem and its mathematical definition. A graph is denoted with n nodes and m edges as a 2-tuple $\mathcal{G} = \{P; E\}$, where P is the node set specified by $P = [p_1; \cdots; p_n] \in R^{d_p \times n}$; E is the edge set $E = [e_1; \cdots; e_m] \in R^{d_e \times m}$; and d_p and d_e are the dimensions of node feature p_i and edge feature e_j, respectively. Given a pair of graphs, $\mathcal{G}_1 = \{P_1; E_1\}$ and $G_2 = \{P_2; E_2\}$, n_1, m_1 denote the number of nodes and edges that belong to $\mathcal{G}_1$, respectively, and likewise for n_2, m_2. The goal of GM is to find the optimal node correspondence matrix X, where X has a one-to-one mapping. $X \in \mathbf{P}$, $\mathbf{P}$ is the set of partial permutation matrices

$$\mathbf{P} = \{X | X \in \{0, 1\}^{n_1 \times n_2}, X\mathbf{1}_{n_2} \leq \mathbf{1}_{n_1}, X^T\mathbf{1}_{n_1} = \mathbf{1}_{n_2}\}, \quad (1)$$

where $\mathbf{1}_{n_1}$ and $\mathbf{1}_{n_2}$ are an $n_1 \times 1$ column vector and an $n_2 \times 1$ column vector, respectively.

Traditionally, the GM problem is represented as maximizing a nonconvex cost function, such that the sum of the node and edge compatibility is maximized:

$$J_{gm}(x) = x^T K x, \quad (2)$$

where x is the vectorization of node correspondence matrix X and K is the affinity matrix representing node similarity and edge similarity. The goal of GM is determining the correspondence matrix X.

To represent the various constraints used in such problems, we represent the GM problem as follows:

$$J_{gm}(x) = x^T K x, \quad s.t. \ \psi(x) \quad (3)$$

where $\psi(x)$ represents various constraints on x. In general, different constraints $\psi(x)$ are required to constrain X. For example, the affine constraint is used to make $\sum_{i_1} X_{i_1 i_2} = 1$ and $\sum_{i_2} X_{i_1 i_2} = 1$, which ensures a one-to-one matching [31]. The sparsity constraint minimizes the number of nonzero elements in X, such as $\|x\|_p = 1, 1 \leq p \leq 2$, $x_i \geq 0$ [43]. Because the affine constraint specifies that each feature point has a maximum of one correspondence, the sparsity constraint eventually leads X to a scarcity of possible matches for each node, which makes it possible to determine the optimal solution.

However, it is inevitable that a mismatches exist between two graphs in the correspondence matrix estimated by GM algorithms. By alternately estimating the spatial transformation and the correspondence matrix between two graphs, the correspondence between the two graphs can be gradually improved. That is, by mapping the source graph to the target graph, the accuracy of the correspondence matrix improves because the mapped points of the source graph are closer to those of the target graph. Correspondingly, the spatial

transformation between two graphs can be estimated more accurately based on this improved correspondence relationship. During this process, it is crucial to robustly estimate the spatial transformation when mismatches exist between the two graphs. The central idea of this paper is to introduce a robust transformation model to the GM algorithm and estimate the spatial transformation between two graphs based on point correspondence.

III. ITERATIVE DEFORMABLE GRAPH MATCHING FRAMEWORK

A. ROBUST SPATIAL TRANSFORMATION MODEL

We employ a compactly supported radial basis function (CSRBF)-based transformation model constrained with l_1 and l_2 regularizations [46] for GM to improve the matching accuracy of traditional GM algorithms. First, we briefly introduce the robust spatial transformation model. Let $P = (p_i, i = 1, \ldots, n)^T$ be the $n \times d$ matrix denoting the source point set, $p_i \in R^d$, and let its corresponding target point set $Q = (q_i, i = 1, \ldots, n)^T$ be an $n \times d$ matrix. The general form of the CSRBF-based transformation is defined as

$$f(\mu) = \beta_0 + \sum_{j=1}^{d} \beta_j \mu^j + \sum_{i=1}^{n} \alpha_i \phi(\mu, p_i, c), \quad (4)$$

where $\mu = (\mu^1, \cdots, \mu^d) \in R^d$, $\phi(\mu, p_i, c) = \phi(\frac{\|\mu - p_i\|_2}{c})$, ϕ is a CSRBF with support c, $\boldsymbol{\beta} = [\beta_j, j = 0, \cdots, d]^T$ is a $(d + 1) \times d$ matrix representing affine coefficients, and $\boldsymbol{\alpha} = [\alpha_i, i = 1, \cdots, n]^T$ is an $n \times d$ matrix representing the elastic coefficients to be estimated.

A linear system constrained with l_1 and l_2 regularizations is employed to estimate $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$.

$$E(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \|Q - A\boldsymbol{\theta}\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}\|_1 + \lambda_2 \|\boldsymbol{\beta} - \boldsymbol{I}\|_2^2, \quad (5)$$

where $\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix}$, and A is an $n \times (n + d + 1)$ matrix given by

$$A = \begin{bmatrix} \phi(p_1, p_1, c) & \ldots & \phi(p_1, p_n, c) & 1 & p_1^1 & \ldots & p_1^d \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi(p_n, p_1, c) & \ldots & \phi(p_n, p_n, c) & 1 & p_n^1 & \ldots & p_n^d \end{bmatrix}. \quad (6)$$

$\boldsymbol{I} = [\mathbf{0}_{1 \times d} \quad \boldsymbol{I}_{d \times d}]^T$ is a constant matrix containing a zero row vector $\mathbf{0}_{1 \times d}$ and an identity matrix $\boldsymbol{I}_{d \times d}$; and λ_1 and λ_2 are regularization parameters. Large values of λ_1 and λ_2 respectively limit the range of elasticity and rigidity of the transformation. In general, λ_2 is set such that it is much larger than λ_1 in order to stabilize the affine transformation.

In equation (5), the l_1 and l_2 norms are employed to regularize the elastic coefficients $\boldsymbol{\alpha}$ and affine coefficients $\boldsymbol{\beta}$, respectively. l_1 regularization is employed to make elastic coefficients $\boldsymbol{\alpha}$ robust and sparse. When location errors exist in some landmarks, l_1 regularization is likely to make their corresponding elastic coefficients zero, implying that these location errors will be eliminated by l_1 regularization. In other

words, the elastic part of the transformation function is robust when l_1 regularization is employed.

In contrast, l_2 regularization is used to stabilize affine coefficients $\boldsymbol{\beta}$, which must be stable in the transformation model to ensure that the deformation does not change significantly. The l_2 norm squares the coefficients in the penalty expression, which forces more equal dispersion of the coefficient values. That is, coefficients regularized by l_2 regularization will be nonzero and will not fluctuate on small data changes, which stabilizes its result.

The constrained least squares problem is solved using the fast iterative shrinkage-thresholding algorithm (FISTA) [47], whereby A is split as: $A = [A^{\boldsymbol{\alpha}} \quad A^{\boldsymbol{\beta}}]$, where $A^{\boldsymbol{\alpha}}$ and $A^{\boldsymbol{\beta}}$ are an $n \times n$ matrix and an $n \times (d+1)$ matrix, respectively. The gradient of $\|Q - A\boldsymbol{\theta}\|_2^2$ can be separated into

$$\nabla \|Q - A\boldsymbol{\theta}\|_2^2 = \begin{bmatrix} -2(A^{\boldsymbol{\alpha}})^T(Q - (A^{\boldsymbol{\alpha}}\boldsymbol{\alpha} + A^{\boldsymbol{\beta}}\boldsymbol{\beta})) \\ -2(A^{\boldsymbol{\beta}})^T(Q - (A^{\boldsymbol{\alpha}}\boldsymbol{\alpha} + A^{\boldsymbol{\beta}}\boldsymbol{\beta})) \end{bmatrix}. \quad (7)$$

To solve $\boldsymbol{\theta}$ iteratively [47], we define $y = \boldsymbol{\theta}^t - \frac{1}{\iota_1}\nabla\|Q - A\boldsymbol{\theta}^t\|_2^2$, where t is the iteration number and ι_1 is the Lipschitz constant of $\nabla\|Q - A\boldsymbol{\theta}\|_2^2$, $\iota_1 = 2\lambda_{max}$, where λ_{max} is the maximum eigenvalue of $A^T A$. We separate y as $y = \begin{bmatrix} y^{\boldsymbol{\alpha}} \\ y^{\boldsymbol{\beta}} \end{bmatrix}$, where $y^{\boldsymbol{\alpha}}$ and $y^{\boldsymbol{\beta}}$ represent the first n and the last $d+1$ entries of y, respectively. The iterative update of $\boldsymbol{\alpha}^{t+1}$ is computed with a soft thresholding operator, as follows:

$$\boldsymbol{\alpha}^{t+1} = \mathcal{S}_{\lambda_1/\iota_1}(y^{\boldsymbol{\alpha}}) \doteq (\cdots, s_{\lambda_1/\iota_1}(y_i^{\boldsymbol{\alpha}}), \cdots), \quad (8)$$

where $s_{\lambda_1/\iota_1}(y_i^{\boldsymbol{\alpha}}) = \text{sgn}(y_i^{\boldsymbol{\alpha}}) \max(|y_i^{\boldsymbol{\alpha}}| - \frac{\lambda_1}{\iota_1}, 0)$ and $y_i^{\boldsymbol{\alpha}}$ is the ith component of $y^{\boldsymbol{\alpha}}$. The iterative update of $\boldsymbol{\beta}^{t+1}$ is given by

$$\boldsymbol{\beta}^{t+1} = \frac{\iota_1 y^{\boldsymbol{\beta}} + 2\lambda_2(\boldsymbol{I})}{\iota_1 + 2\lambda_2}. \quad (9)$$

The most important advantage of the CSRBF-based transformation model constrained with l_1 and l_2 regularizations is that it is robust to the location errors of corresponding point pairs. When mismatched point pairs exist in the corresponding point pair set, the constrained transformation model achieves a reasonable deformation field compared to traditional transformation models. Fig. 1 shows an example of deformation fields using different transformation models. In this example, there are mismatches in the upper-right corner where the black squares are source points, the black dots are target points, and the green arrow is the correspondence between points in Fig. 1(a). Fig. 1(b) shows the TPS deformation result when the black grid is distorted and the deformation field cannot preserve topology information. However, the deformation field is smoothed using our transformation model with l_1 and l_2 norms because it is robust to outliers in the corresponding point pairs in Fig. 1(c).

B. ITERATIVE DEFORMABLE GRAPH MATCHING

For two given graphs, $\mathcal{G}_1 = \{P_1; E_1\}$ and $G_2 = \{P_2; E_2\}$, suppose that a spatial transformation between $\mathcal{G}_1$ and $\mathcal{G}_2$ is denoted by f. The aim of deformable GM is to render it
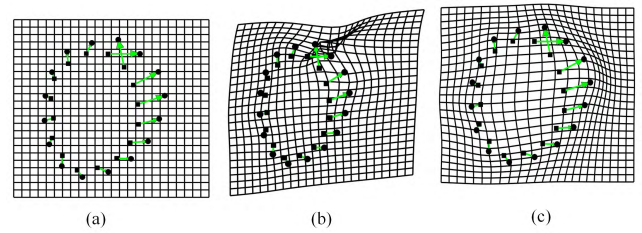

FIGURE 1. Example of transformations: (a) the source and target points are marked by black squares and dots, respectively, and correspondence relationships are marked by green arrows. Note the mismatches in the upper-right corner. (b) and (c) Show the deformation fields estimated by TPS and the CSRBF-based transformation model constrained with l_1 and l_2 regularizations, respectively.

more robust to the deformation between two graphs. That is, in deformable GM, correspondence matrix X and the spatial transformation are estimated at the same time. This idea is similar to that of point set matching [48], [49]. Then, the cost function of deformable GM can be represented as

$$\min_{f,x} J(\boldsymbol{\alpha}, \boldsymbol{\beta}, x) = -x^T K(f)x + \lambda_1 \|\boldsymbol{\alpha}\|_1 + \lambda_2 \|\boldsymbol{\beta} - I\|_2^2,$$
$$s.t. \ \psi(x) \quad (10)$$

where f is the spatial transformation between two graphs, which is determined by $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$; $K(f)$ is the affinity matrix between the mapped source graph and the target graph; x is the vectorized X; and $\psi(x)$ represents various constraints on vector x.

Affinity matrix $K(f)$ is composed of the node affinity on its diagonal and the edge affinity on its off-diagonals. The elements of the node affinity and the elements of the edge affinity are computed as follows:

$$k_{i_1,i_2} = -\|p_{i_2} - f(p_{i_1})\|_2^2,$$
$$k_{e_1,e_2} = -\|(p_{i_2} - p_{j_2}) - (f(p_{i_1}) - f(p_{j_1}))\|_2^2, \quad (11)$$

where p_{i_1} and p_{i_2} are nodes of $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively; p_{i_1} and p_{j_1} are the two nodes of edge e_1 in $\mathcal{G}_1$, and p_{i_2} and p_{j_2} are the two nodes of edge e_2 in $\mathcal{G}_2$.

We alternately estimate transformation f and point correspondence matrix X. First, the point correspondence can be estimated using any GM algorithm. Next, the spatial transformation is estimated based on point correspondence X. After the spatial transformation is estimated, affinity matrix $K(f)$ is recomputed and the GM algorithm is applied again to update point correspondence matrix X. Next, the parameters of the spatial transformation are estimated based on update X. The above process is repeated several times until X no longer changes. The iterative algorithm for the optimizing model in Eq. (10) is summarized in Algorithm 1.

We employ the CSRBF-based transformation model constrained with l_1 and l_2 regularizations [46] as f. Radial basis functions $\phi(r/c)$ are used with compact support c, as in the Wendland functions [50]. Support c can be modified to adjust for the influence of landmarks. Then, based on Eq. (4),

---

Algorithm 1 Iterative Deformable Graph Matching

Require: Extract the point sets from the source and target image, respectively, to be the source and target point sets.

Initialize: λ_1, λ_2, the threshold η.

while the stopping condition is unsatisfied **do**

    1) Construct the target graph $\mathcal{G}_2$.

    2) Construct the source graph $\mathcal{G}_1$.

    3) Estimate the correspondence matrix X using a GM algorithm.

    4) Estimate the transformation parameters $\boldsymbol{\theta}$ using Algorithm 2 based on X. Decrease λ_1, λ_2.

    5) Map the source point set using the transformation with parameters $\boldsymbol{\theta}$.

    6) When the average distance between the mapped source points in two iterations is smaller than η, then stop; otherwise, go to Step 2.

end while

$f(p_{i_1})$ can be rewritten in a linear format:

$$f(p_{i_1}) = \mathbf{A}_{i_1}\theta, \tag{12}$$

where $\mathbf{A}_{i_1}$ is a row vector, which is the $i_1{}^{th}$ line of matrix A, by substituting p_i with p_{i_1}, $i_1 = 1, \ldots, n_1$ in Eq. (6). Then,

$$f(p_{i_1}) - f(p_{j_1}) = (\mathbf{A}_{i_1} - \mathbf{A}_{j_1})\boldsymbol{\theta}. \tag{13}$$

We denote $(\mathbf{A}_{i_1} - \mathbf{A}_{j_1})$ by $\mathbf{A}_{i_1 j_1}$. Suppose that point correspondence matrix X is known; then, Eq. (13) can be simplified:

$$f(p_{i_1}) - f(p_{j_1}) = (\mathbf{A}_{i_1 j_1})\boldsymbol{\theta}. \tag{14}$$

Similarly, $p_{i_2} - p_{j_2}$ is denoted by $p_{i_2 j_2}$.

In Eq. (11), $K(f)$ is represented as $K_p(f) + K_e(f)$, where $K_p(f)$ and $K_e(f)$ contain only diagonal and off-diagonal elements of $K(f)$, respectively. Then,

$$-x^T K_p(f)x = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} x_{i_1 i_2} \|p_{i_2} - \mathbf{A}_{i_1}\theta\|_2^2, \tag{15}$$

and

$$-x^T K_e(f)x = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} x_{i_1 i_2} x_{j_1 j_2} \|p_{i_2 j_2} - \mathbf{A}_{i_1 j_1}\theta\|_2^2 \tag{16}$$

Equation (10) is rewritten as

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta},x} J(\boldsymbol{\alpha}, \boldsymbol{\beta}, x) = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} x_{i_1 i_2} \|p_{i_2} - \mathbf{A}_{i_1}\theta\|_2^2$$

$$+ \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} x_{i_1 i_2} x_{j_1 j_2} \|p_{i_2 j_2} - \mathbf{A}_{i_1 j_1}\theta\|_2^2$$

$$+ \lambda_1 \|\boldsymbol{\alpha}\|_1 + \lambda_2 \|\boldsymbol{\beta} - I\|_2^2,$$

$$s.t. \ \psi(x) \tag{17}$$

Let

$$h(\boldsymbol{\theta}) \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} x_{i_1 i_2} \|p_{i_2}\mathbf{A}_{i_1}\boldsymbol{\theta}\|_2^2$$

$$+ \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} x_{i_1 i_2} x_{j_1 j_2} \|p_{i_2 j_2}\mathbf{A}_{i_1 j_1}\boldsymbol{\theta}\|_2^2. \tag{18}$$

Then, the gradient of $h(\boldsymbol{\theta})$ is

$$\nabla h(\boldsymbol{\theta}) = -2 \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} x_{i_1 i_2} \left[\mathbf{A}_{i_1}^T \left(p_{i_2}\mathbf{A}_{i_1}\boldsymbol{\theta}\right)\right]$$

$$-2 \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} x_{i_1 i_2} x_{j_1 j_2} \left[\mathbf{A}_{i_1 j_1}^T \left(p_{i_2 j_2}\mathbf{A}_{i_1 j_1}\boldsymbol{\theta}\right)\right]. \tag{19}$$

As mentioned above, we define $y = \boldsymbol{\theta}^t - \frac{1}{\iota}\nabla h(\boldsymbol{\theta}^t)$, ι is the Lipschitz constant of $\nabla h(\boldsymbol{\theta})$. Further, we separate y as $y = \begin{bmatrix} y^{\boldsymbol{\alpha}} \\ y^{\boldsymbol{\beta}} \end{bmatrix}$. Elastic coefficient $\boldsymbol{\alpha}$ and affine coefficient $\boldsymbol{\beta}$ are solved using Eqs. (8) and (9), respectively. The procedure for estimating the parameters of the spatial transformation is incorporated into Algorithm 2.

Algorithm 2 Solving the Transformation Parameter

Input:

    Compute a Lipschitz constant ι of $\nabla h(\boldsymbol{\theta})$ and set λ_1, λ_2, $s^1 = \boldsymbol{\theta}^0$, $\rho^1 = 1$, $t = 1$;

Output:

    Coefficients of the transformation function $\boldsymbol{\theta}$;

1: **while** the stopping condition is unsatisfied **do**

2:     Compute $\nabla h(s^t)$ using (19);

3:     Compute $y = s^t - \frac{1}{\iota}\nabla h(s^t)$;

4:     Estimate $\boldsymbol{\alpha}^t = \mathcal{S}_{\lambda_1/\iota}(y^{\boldsymbol{\alpha}})$ using (8);

5:     Estimate $\boldsymbol{\beta}^t = \frac{\iota y^{\boldsymbol{\beta}} + 2\lambda_2 \boldsymbol{I}}{\iota + 2\lambda_2}$;

6:     Compute $\boldsymbol{\theta}^t = \begin{bmatrix} \boldsymbol{\alpha}^t \\ \boldsymbol{\beta}^t \end{bmatrix}$;

7:     Compute $\rho^{t+1} = \frac{1+\sqrt{1+4(\rho^t)^2}}{2}$;

8:     Compute $s^{t+1} = \boldsymbol{\theta}^t + \left(\frac{\rho^t - 1}{\rho^{t+1}}\right)(\boldsymbol{\theta}^t - \boldsymbol{\theta}^{t-1})$ and $t = t+1$;

9: **end while**

IV. CARDIAC MOTION ESTIMATION USING MULTILAYER ITERATIVE DEFORMABLE GRAPH MATCHING

A. LV GRAPH

The goal of cardiac motion estimation is to estimate the spatial transformation between two cardiac images at different time points, such as the ES and ED phases. We employ iterative deformable GM to estimate the correspondence and spatial transformation between LV contours at different time points. First, the contours of the LV myocardium are extracted and LV graphs are constructed. Here, we employ a convolutional neural network (CNN) [51] to segment the LV endocardium and epicardium.
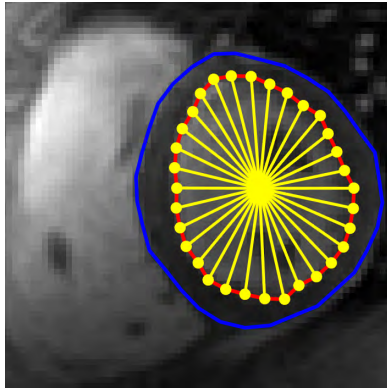
FIGURE 2. Illustration of an LV graph. The red and blue lines mark the contours of the endocardium and epicardium, respectively. The nodes and edges of the endocardium graph are indicated by yellow points and lines, respectively.

Considering that the LV slice along the short-axis is circular, we employ [52] to concisely construct the LV graph. Our LV graph method differs from traditional graph construction algorithms, such as full-link and Delaunay triangulation. Full-link and Delaunay triangulation algorithms introduce too many edges without clear meaning, whereas our LV graph is constructed utilizing the anatomical structure of the LV. An auxiliary point located in the center of the LV is introduced, and graph edges are defined as the lines connecting the nodes to the auxiliary point, as shown in Fig. 2. The figure shows a graph of the endocardium; the epicardium graph is constructed in the same manner. In our LV graph, each edge refers to the radius of the LV, and only limited edges exist in the graph, simplifying the GM process.

Fig. 3 depicts two synthetic LV graphs, where $\mathcal{G}_1$ contains four nodes and three edges, and $\mathcal{G}_2$ contains three nodes and two edges. Note that central points 4 and c are the auxiliary nodes for the respective graphs and X is the correspondence matrix. Affinity matrix K contains the correspondence of nodes and edges. In K, the diagonal and off-diagonal elements encode the similarity between nodes and edges respectively, where zeros indicate no correspondence and nonzero values represent similarity. Specifically, $(\cdot, \cdot)$ refers to the Euclidean distance between two nodes. For example, $(1, a)$ refers to the Euclidean distance between nodes 1 and a. Because the auxiliary nodes of two graphs must be matched by translation, k_{e_1, e_2} can be simplified to $-\|p_{i_2} - f(p_{i_1})\|_2^2$, as shown in Fig. 3(b).

B. MULTILAYER FRAMEWORK

As many points are extracted from the contours of the LV myocardium, it is difficult to distinguish those that are close to each other. To alleviate the interaction between close points, we propose a multilayer iterative deformable GM framework (MIDGM). In the proposed framework, sparse points are selected uniformly from all points to construct a sparse LV graph and iterative deformable GM is performed to estimate the transformation between sparse
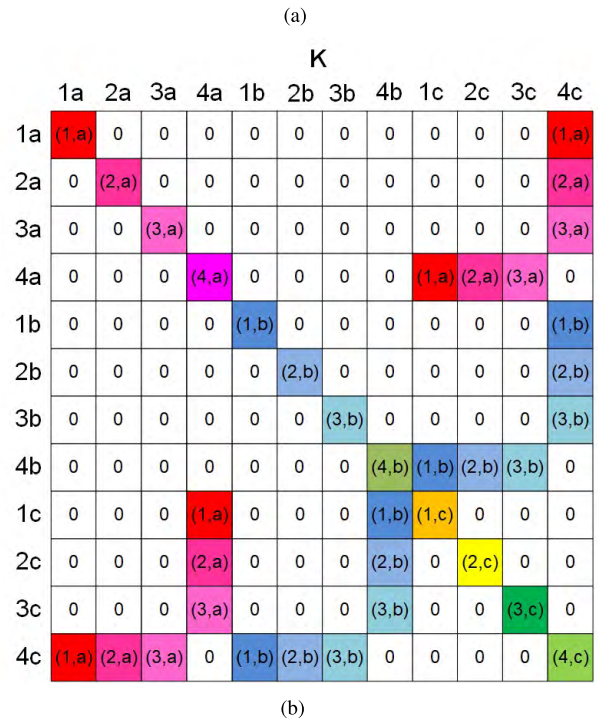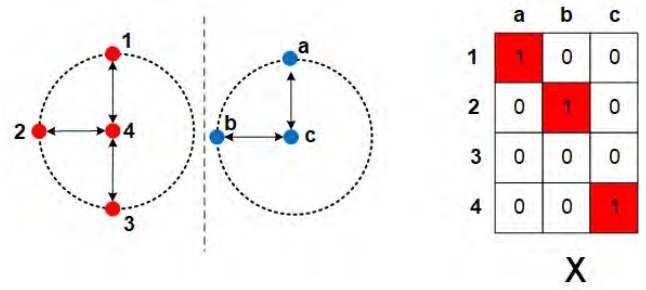

FIGURE 3. Example of LV graph matching: (a) two synthetic graphs and the correspondence matrix X; (b) the affinity matrix K.

points. Next, all source points are mapped using the estimated transformation and dense points are selected from all points to perform iterative deformable GM again to estimate a new transformation. The above process is repeated until all points are used to estimate the transformation between two graphs. Fig. 4 illustrates the process of the multilayer framework. During this process, when a small number of points are selected to estimate the transformation between two graphs, sparse points are easily distinguished because they are far apart, which results in robust and coarse transformation estimations. Note that we cannot ensure that the source points correspond one-to-one to the target points. However, our goal is to select the most similar point from the point set. As illustrated in Fig. 5(a) and (b), points 1, 2, and 3 are extracted source points, and points a, b, and c are extracted target points. Source point 2 may correspond to point a, b, or c. Next, the goal of our proposed method is to determine which of these points is the most similar point. This process employs not only the node features, but also the edge features.
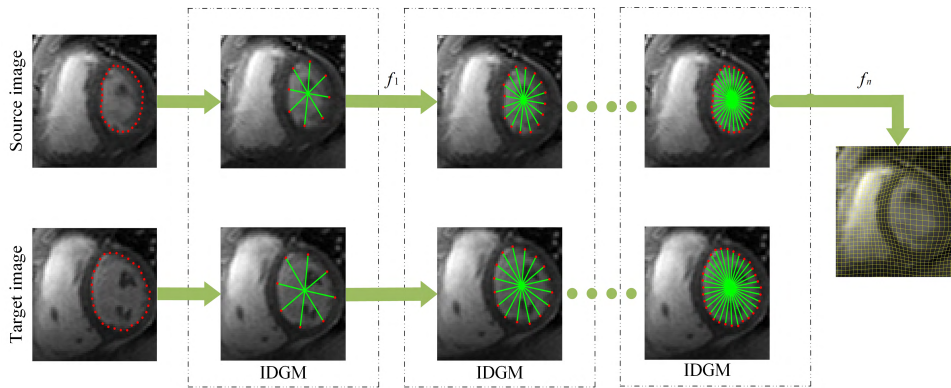
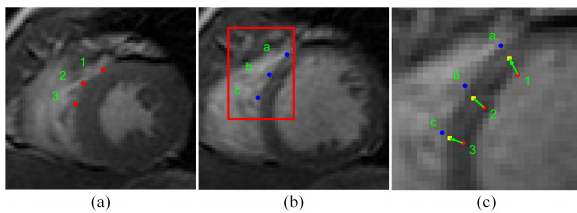**FIGURE 4.** Multilayer iterative deformable graph matching framework.

FIGURE 5. (a) Extracted source points from the epicardium at the ES phase. (b) Extracted target points from the epicardium at the ED phase. (c) The yellow square points are mapped source points, and green arrows indicate the displacement vectors. The background is the enlarged local region of (b).

In the next step, the source points are mapped using the initial transformation and all corresponding points are closer to each other than the original corresponding points, making it easier to estimate the correspondence between the mapped source and target point sets. Although it is impossible to ensure that the extracted sparse points correspond to each other in features, it is nevertheless feasible to select one point from the sparse target point set that is most similar to the given source point. Even if the transformation is not accurate, the source points are mapped to the target points more closely, making it easier to select corresponding points because the mapped points are closer to the target points than are the source points to the target points. Thus, the transformation is not required to be accurate, but it is required to provide a coarse displacement, as illustrated in Fig.5(c). Our multilayer framework estimates point correspondences on a layerwise basis, from the bottom to the top. As the number of points increases, the transformation becomes progressively more accurate. Ultimately, an accurate transformation can be obtained.

Fig. 6 shows an example that demonstrates the validity of our proposed MIDGM framework. Here, factorized GM (FGM) [32] is employed to estimate the correspondence between nodes, denoted by MIDGM + FGM. Discrete solution X is obtained by employing the Hungarian algorithm on continuous matrix X. The performance of MIDGM + FGM is compared to that of deformable GM (DGM) [45]. We emphasize that FGM is also

performed in DGM to estimate node correspondences. The differences between MIDGM + FGM and DGM are twofold: first, MIDGM uses the sparsity-constrained transformation model, whereas DGM adopts the traditional radial basis function interpolation model constrained by bending energy (BE); second, MIDGM employs a multilayer framework, whereas DGM is performed on only a single layer. The source and target LV images and extracted point sets are illustrated in Fig. 6(a). The correspondence accuracy estimated by MIDGM for each layer is shown in Fig. 6(b). Note that the correspondence accuracy in Layer 1 is very high, validating the advantages of MIDGM. Moreover, high correspondence accuracy is maintained with the bottom-up procedure. Fig. 6(c) shows the correspondence accuracy estimated by MIDGM (in the last layer) and DGM at different iterations. The correspondence accuracy gradually improves with iterative deformable GM, whether by MIDGM or DGM, demonstrating the effectiveness of alternating estimation of the correspondence and spatial transformation. After several iterations, the correspondence accuracy gradually converges. MIDGM clearly outperforms DGM in terms of the correspondence accuracy throughout the iterations. This difference is conspicuous in the first iteration, where the correspondence accuracy of MIDGM is better than that of DGM, demonstrating the advantages of our multilayer framework, with which only sparse nodes are used to estimate correspondence. The deformation fields estimated by MIDGM and DGM in the first four iterations are illustrated in Fig. 6(d). We emphasize that DGM is sensitive to mismatched points because a traditional radial basis function interpolation model is employed, which easily results in non-topology-preserving deformation fields. To preserve the topology of the deformation fields, a large penalty coefficient is used for the BE in the radial basis function interpolation model, resulting in deformation similar to affine deformations, as shown in the second row of Fig. 6(d). The first row of Fig. 6(d) shows that the deformation fields estimated by MIDGM are smoother and more reasonable than those estimated by DGM, owing to its high correspondence accuracy and the robust spatial transformation model. Indeed, DGM could not suppress the influence of
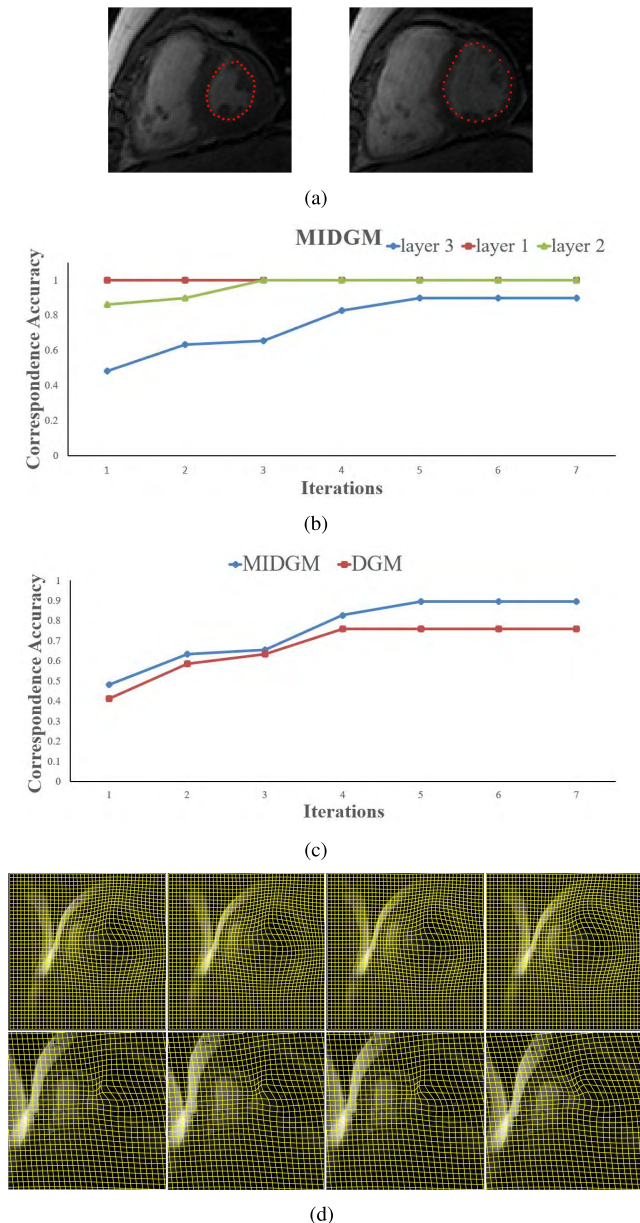
FIGURE 6. Comparison of the correspondence accuracy and deformation fields estimated by MIDGM and DGM: (a) source and target images; (b) correspondence accuracy obtained by MIDGM in different layers; (c) comparison of correspondence accuracy using MIDGM (in the last layer) and DGM over different iterations; (d) deformation fields estimated by MIDGM and DGM in the first four iterations. The first and second rows show results using MIDGM and DGM, respectively.

mismatched points and estimates deformation fields similar to affine transformations that are unreasonable for cardiac motion.

V. COMPUTATION TIME

As the multilayer framework is employed in the proposed model, it is necessary to discuss the computation time of our proposed method. The main computation cost in our method depends on the computation cost of the robust transformation and the number of layers of the multilayer framework. Here, we must emphasize that the fast iterative

shrinkage-thresholding algorithm (FISTA) is employed to estimate the coefficients in the robust transformation. Thus, the computation cost of the robust transformation is relatively higher than that of the traditional RBF-based transformation. To facilitate comparison of the computation times of different algorithms, the respective computation times of traditional GM algorithms, including the spectral matching with affine constraints (SMAC) [31], sparse nonnegative matrix factorization graph matching (PSM) [43], and factorized graph matching (FGM) [32], deformable GM (DGM) [45], and our proposed method are provided in Table 1. In this experiment, an LV image at the ES phase and an LV image at the ED phase are matched with each other. The number of source points and target points were 30, respectively, including two outliers. Three layers were employed in our method, with 8, 15, and 30 points in the first, second, and third layers, respectively. This experiment was performed on a computer with a 3.3-GHz quad-core processor and 12 GB RAM, using MATLAB 2015a. In Table 1, the computation time of each layer in our proposed method is also provided. It can be seen that the computation time of our proposed method is directly related to the GM algorithm used in our method. For example, the computation time of FGM is larger than that of PSM; correspondingly, the computation time of MIDGM + FGM is larger than that of MIDGM + PSM. Compared with DGM, considering that our method is performed three times, the average computation time of each layer in our method is less than that of DGM.

VI. EXPERIMENTS

This section provides experimental results on two benchmark datasets that we used to evaluate the performance of MIDGM. MIDGM was compared to several state-of-the-art GM algorithms. We performed two experiments. The first experiment compared MIDGM to several state-of-the-art GM algorithms with respect to the correspondence accuracy. The second experiment involved comparing LV motion estimation using MIDGM, the GM algorithms, and state-of-the-art algorithms on the same datasets.

A. CINE MRI CARDIAC DATASETS

Two cine MR cardiac image datasets were employed to evaluate the performance of the MIDGM and GM algorithms. The first consisted of cardiac MRI sequences acquired from 33 subjects, provided by the Department of Diagnostic Imaging at the Hospital for Sick Children (HSC) in Toronto, Canada [53]. Each subject's image sequence consisted of 20 frames and the number of slices acquired along the long axis of the subject ranged from 8 to 15. The distance between slices ranged from 6 to 13 *mm*. The size of all image slices was 256×256 pixels with pixel-spacing of $0.93-1.64$ *mm*. The endocardial and epicardial contours of each image were manually delineated to provide a ground truth.

The second dataset was the MICCAI2009 challenge dataset provided by Sunnybrook Health Sciences Centre,

TABLE 1. Comparison of the computation time using our method and state-of-the-art methods for matching two LV images at the ES and ED phases. A hyphen ("—") denotes no-operation using state-of-the-art methods.

Time(s)	Methods						
	PSM	MIDGM + PSM	SMAC	MIDGM + SMAC	FGM	MIDGM + FGM	DGM
Layer 1	—	0.23	—	0.08	—	1.80	—
Layer 2	—	0.49	—	0.14	—	4.38	—
Final result	0.61	12.80	0.26	8.82	1.33	25.19	10.67

also located in Toronto, Canada [54]. The dataset consisted of 45 short-axis MRI cases from a range of patients and pathologies. These cases were randomly divided into three groups: 15 cases for training, 15 cases for testing, and 15 cases for an online contest. Moreover, manual segmentation was provided by an experienced cardiologist for the ED and ES phases in each case. These manual segmentation results were used as the ground truth for the evaluation.

B. CORRESPONDENCE ACCURACY

This section describes our evaluation of MIDGM with respect to the correspondence accuracy. We randomly selected 17 subjects from the HSC dataset and employed images at the ES phase as the source images and images at the ED phase as the target images. The corresponding slices of the source and target images were used for matching. Each endocardial and epicardial contour of a slice consisted of 32 landmarks. As each contour was drawn starting from the junction of the left and right ventricles, located closest to the posterior interventricular sulcus, these 32 landmarks corresponded to one another and could be used to evaluate the corresponding accuracy of the different GM algorithms. Moreover, 32 landmarks from the myocardium, delineated by experts, were used to construct graphs. Endocardial and epicardial contours were used to construct endocardial and epicardial graphs, respectively, and the proposed MIDGM framework was implemented separately on each graph type.

We compared our proposed MIDGM to three state-of-the-art GM algorithms: SMAC, PSM, and FGM. We used these methods to estimate the correspondence between endocardial points at the ES and ED phases. (The correspondence between epicardial points at the ES and ED phases was estimated in a similar manner.) With MIDGM, various GM algorithms were employed to estimate the correspondence matrix. We denote these as MIDGM + GM: for example, MIDGM + SMAC which means that we performed MIDGM by employing SMAC to estimate correspondences. In these experiments, DGM was not performed because it easily results in non-topology-preserving deformations, as explained above.

The average accuracy (ACC) was used to measure the correspondence accuracy:

$$Acc = tr(X^T X_{tru})/tr(\mathbf{1}_{n_2 \times n_1} X_{tru}), \quad (20)$$

where X_{tru} is the ground truth of the correspondence matrix and $\mathbf{1}_{n_2 \times n_1}$ is an $n_2 \times n_1$ matrix in which all elements are 1.

Acc computes consistent matches between estimated correspondence matrix X and ground truth X_{tru}. $Acc = 1$ indicates an exact match. For each subject, the correspondence accuracy of all slices was averaged as the subject's matching accuracy.

Furthermore, the objective score (SCO), $SCO = \max_x x^T K(f)x$, was computed to evaluate the GM performance of MIDGM and the other GM algorithms.

We also evaluated our method with outlier nodes in both $\mathcal{G}_1$ and $\mathcal{G}_2$. To do so, several points were eliminated randomly from the two graphs, and then MIDGM, SMAC, PSM, and FGM were performed to estimate the correspondence between these points. There were 0−−5 outliers in each of the two graphs, resulting in 0−−10 mismatched point pairs. Table 2 shows the average ACC and SCO using MIDGM and three GM algorithms with a varying number of outliers. MIDGM considerably outperformed the other algorithms with respect to both the ACC and SCO, especially in cases with many mismatched points (e.g., when $m = 5$).

Fig. 7 illustrates the correspondence between two LV contours using different algorithms. The yellow lines indicate correct matches and the red lines denote mismatches. Note that fewer mismatched pairs were obtained by MIDGM compared to the other GM algorithms.

C. LV MOTION ESTIMATION

To further verify the validity of MIDGM for LV motion estimation, we automatically extracted LV contours from cine MRIs and estimated the spatial transformation between LV contours at the ES and ED phases. Images from the remaining 16 subjects of the HSC dataset and 30 cases (15 test cases and 15 online cases) from the MICCAI2009 challenge dataset were employed to evaluate the performance of our proposed MIDGM. With MIDGM, we can estimate the transformation between LV contours at the ES and ED phases. For each subject or case, the ES and ED images (4−−10 slices) were matched slice-by-slice. Furthermore, the contours marked by experts in the ES phase were mapped to those in the ED phase using the estimated transformation. Thus, the contours of the LV in the ED phase were automatically segmented. The expert-segmented contours at the ED phase were employed to evaluate the accuracy of LV motion estimation.

To extract LV endocardial and epicardial contours, we employed a fully convolutional neural network (FCN) [51]. The dataset images at two given phases from among the HSC dataset that did not contain images from the ES and

TABLE 2. Average correspondence accuracy and objective score of MIDGM and three state-of-the-art graph matching algorithms, with optimal results in bold.

Outliers number m	ACC						SCO					
	PSM	MIDGM +PSM	SMAC	MIDGM +SMAC	FGM	MIDGM +FGM	PSM	MIDGM +PSM	SMAC	MIDGM +SMAC	FGM	MIDGM +FGM
$m = 0$	0.59	**0.85**	0.54	**0.76**	0.89	**0.99**	61.08	**77.39**	60.41	**76.83**	62.73	**74.69**
$m = 1$	0.53	**0.80**	0.52	**0.73**	0.75	**0.86**	59.02	**74.19**	58.38	**73.55**	60.47	**75.23**
$m = 2$	0.50	**0.75**	0.49	**0.69**	0.66	**0.75**	57.34	**70.94**	56.39	**70.11**	58.51	**72.60**
$m = 3$	0.48	**0.69**	0.48	**0.65**	0.60	**0.69**	55.28	**67.57**	54.41	**66.89**	56.39	**69.43**
$m = 4$	0.46	**0.66**	0.46	**0.61**	0.56	**0.67**	53.22	**64.46**	52.16	**63.97**	54.15	**66.31**
$m = 5$	0.46	**0.64**	0.45	**0.61**	0.56	**0.66**	50.79	**61.30**	49.82	**60.72**	51.76	**63.02**
Mean	0.50	**0.73**	0.50	**0.68**	0.67	**0.77**	56.12	**69.31**	55.26	**68.68**	57.33	**70.21**
Std	0.05	0.08	0.03	0.06	0.13	0.13	3.80	6.04	3.94	6.02	4.06	4.87

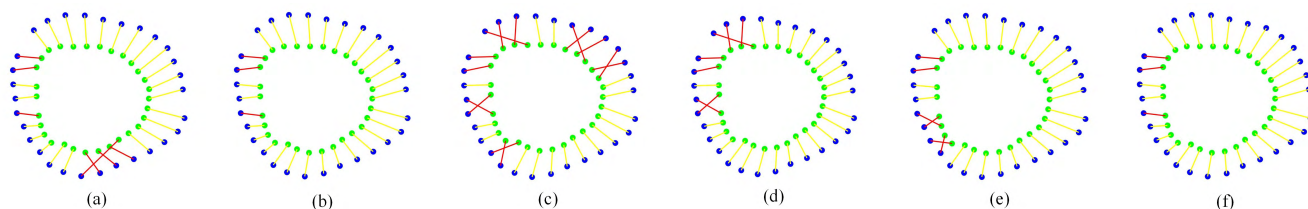(a)   (b)   (c)   (d)   (e)   (f)

FIGURE 7. Example of LV contour matching using different algorithms when $m = 2$. The yellow and red lines indicate correct matches and mismatches, respectively. The results of PSM, MIDGM + PSM, SMAC, MIDGM + SMAC, FGM and MIDGM + FGM are shown from left to right.

ED phases, and all training cases in the MICCAI2009 dataset were used to train the FCN. The trained FCN was used to extract the ED and ES contours, and evenly spaced points were extracted from the ES and ED contours to serve as the source and target nodes, respectively. After constructing the LV graphs, we matched the ES graph to the corresponding ED graph using MIDGM, SMAC, PSM, and FGM, respectively. We then estimated the transformations between two LV contours. Insofar as only the endocardial contours in the ES phase were provided in the MICCAI2009 challenge dataset, we used only the endocardial contours provided by experts as the ground truth when evaluating the algorithms.

The average symmetric contour distance (SCD) [55], average perpendicular distance (APD) [54], BE, and shear strain (SST) [56] were used to evaluate the estimated deformation fields. The SCD computes the mean symmetric contour distance to the ground-truth contours weighted by the number of landmarks in each object. The APD measures the distance from the automatically segmented contour to the corresponding manually drawn expert contour, averaged over all contour points. The SST measures the deformation displacement field of an image along each axis, which is equal to the length of the deformation at its maximum divided by the perpendicular length in the plane of force application. The BE represents the smoothness of the deformation fields, defined as

$$BE = \int \int \left[\left(\frac{\partial^2 f}{\partial x^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 f}{\partial y^2}\right)^2 \right] dxdy \quad (21)$$

in $2-d$ spatial space.

1) HSC DATASET

Table 3 shows the various evaluation indicators for 16 cases in the HSC dataset using different algorithms. Compared to PSM, SMAC, and FGM, MIDGM returned the optimal SCD, BE, and SST, owing to its high correspondence accuracy and robust transformation. Non-topology-preserving deformation fields occurred in the PSM and SMAC results (in Cases 24 and 30), whereas MIDGM + PSM and MIDGM + SMAC preserved the topology of deformation fields due to their robust transformation model. Note that the BE and SST of the deformation fields estimated by MIDGM were significantly smaller than those of other GM algorithms. This result is consistent with the regularizations used in the transformation model and clearly demonstrates that MIDGM can generate smoother deformation fields for LV motion estimation.

To illustrate the matching results in detail, Fig.8 shows the displacement vectors using MIDGM, PSM, SMAC, and FGM. The source image is the seventh slice from Subject 19 in the HSC dataset. The target image is its corresponding slice at the ED phase. The first row shows the LV displacement vectors from the ES to ED phase using PSM, SMAC, and FGM, where blue and red points represent the source and target points, respectively, and green arrows illustrate the displacement vectors of these points. The second row shows the results using the corresponding MIDGM + GM algorithms. The mapped source points are expected to be close to the target points. Note that the displacement vectors estimated by SMAC and FGM are more slanted than the

TABLE 3. Measurements of LV motion estimation using MIDGM, PSM, SMAC, and FGM, with optimal results in bold. (A hyphen ("-") denotes a non-topology-preserving deformation field. Note: SST (*value* $\times 10^{-4}$), SCD (*mm*).)

| Cases	PSM			MIDGM+PSM			SMAC			MIDGM+SMAC			FGM			MIDGM+FGM		
	SCD	BE	SST	SCD	BE	SST	SCD	BE	SST	SCD	BE	SST	SCD	BE	SST	SCD	BE	SST
18	1.74	19.28	5.41	**1.38**	**15.47**	**4.84**	2.13	12.30	4.05	**1.42**	**8.94**	**2.94**	1.78	16.58	5.40	**1.33**	**12.69**	**4.15**
19	2.63	14.52	5.19	**1.00**	**9.22**	**3.46**	1.66	7.37	3.85	**1.11**	**3.24**	**1.90**	0.97	8.63	3.53	**0.90**	**5.52**	**2.34**
20	2.00	2.05	2.48	**1.57**	**1.83**	**1.80**	2.00	3.77	2.67	**1.95**	**3.36**	**2.44**	1.74	2.36	2.57	**1.53**	**1.00**	**1.39**
21	2.52	31.47	6.16	**1.97**	**24.67**	**5.78**	-	-	-	**2.73**	**35.59**	**6.54**	2.40	23.05	5.71	**1.84**	**17.98**	**4.52**
22	3.20	12.03	3.59	**2.46**	**6.11**	**3.47**	3.45	12.72	4.64	**3.08**	**10.69**	**3.93**	2.54	12.02	4.80	**2.48**	**8.18**	**3.25**
23	3.30	18.67	9.22	**2.95**	**14.20**	**5.53**	3.06	19.17	5.37	**2.78**	**11.08**	**4.58**	2.83	13.69	6.61	**2.64**	**4.60**	**3.00**
24	-	-	-	**3.64**	**18.41**	**6.32**	2.85	20.42	6.48	**2.53**	**13.89**	**4.47**	2.61	17.06	5.80	**2.30**	**12.84**	**4.23**
25	3.50	15.79	4.38	**2.33**	**13.75**	**3.87**	3.44	7.44	3.26	**3.05**	**6.88**	**3.04**	2.66	7.25	3.42	**2.32**	**6.12**	**2.80**
26	3.93	17.94	4.04	**3.07**	**14.86**	**3.96**	4.07	10.31	3.70	**3.28**	**9.47**	**3.44**	3.08	13.76	4.35	**2.43**	**11.12**	**3.54**
27	-	-	-	**3.25**	**5.67**	**2.75**	3.72	12.36	5.28	**3.33**	**2.74**	**2.03**	3.01	10.14	5.03	**2.79**	**3.66**	**2.48**
28	4.46	17.56	5.16	**2.36**	**10.08**	**4.08**	3.57	9.22	4.05	**2.49**	**8.23**	**3.18**	3.37	12.15	4.66	**2.16**	**9.22**	**3.78**
29	3.82	11.26	3.93	**2.71**	**8.95**	**3.57**	2.64	12.83	6.19	**1.72**	**7.39**	**4.17**	3.41	9.26	3.96	**2.72**	**5.95**	**2.67**
30	-	-	-	**2.58**	**15.64**	**6.47**	-	-	-	**2.23**	**13.52**	**4.36**	2.77	12.56	4.56	**1.61**	**8.46**	**3.11**
31	3.86	13.53	3.80	**3.08**	**10.51**	**3.56**	2.95	17.65	4.96	**2.02**	**14.42**	**3.79**	2.76	16.80	4.64	**2.03**	**15.48**	**4.18**
32	1.91	6.78	2.67	**1.39**	**3.29**	**2.54**	1.54	4.95	2.92	**1.39**	**4.51**	**2.75**	1.44	4.21	2.89	**1.36**	**3.64**	**2.50**
33	1.99	7.27	3.50	**1.71**	**5.24**	**2.86**	2.43	9.34	4.74	**1.38**	**6.43**	**2.79**	1.33	5.20	3.30	**1.24**	**4.18**	**1.20**
Mean	2.99	14.47	4.58	**2.34**	**11.12**	**4.05**	2.82	11.42	4.44	**2.28**	**10.02**	**3.52**	2.42	11.55	4.45	**2.04**	**8.17**	**3.07**
Std	0.91	7.28	1.76	**0.77**	**6.07**	**1.38**	0.79	5.02	1.15	**0.73**	**7.78**	**1.16**	0.74	5.41	1.12	**0.68**	**4.75**	**0.98**
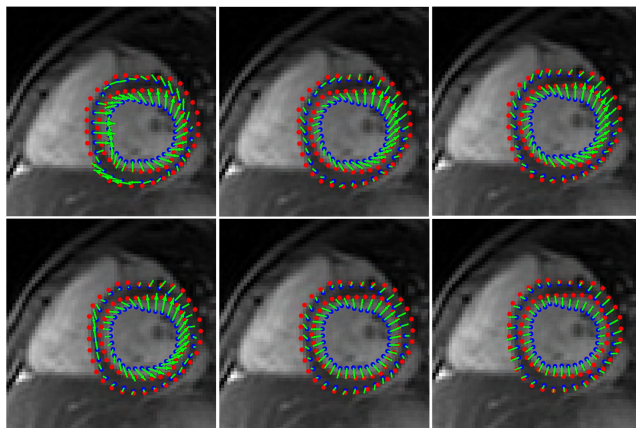
FIGURE 8. Illustration of the displacement vectors using MIDGM, PSM, SMAC and FGM. The first row shows the displacement vectors using PSM, SMAC and FGM, where blue and red points represent the source and the target points, respectively, and green arrows illustrate the displacement vectors of these points. The second row shows results using MIDGM + PSM, MIDGM + SMAC and MIDGM + FGM, respectively.

FIGURE 9. Illustration of the SST using different algorithms. In the first row, from left to right: PSM, SMAC, and FGM. The results using MIDGM + PSM, MIDGM + SMAC, and MIDGM + FGM are shown in the second row.

vectors estimated by MIDGM + SMAC and MIDGM + FGM. The myocardium dilates as it moves from the ES phase to the ED phase, which means that its radial motion is more reasonable and implies that MIDGM + SMAC and MIDGM + FGM outperformed SMAC and FGM. However, the displacement vectors are abnormal for PSM, owing to the mismatches between points. Nevertheless, the displacement vectors can be improved by using MIDGM + PSM, which demonstrates the advantage of the robust transformation model used in our method. Furthermore, Fig. 9 illustrates the SST values for the corresponding deformation fields in Fig.8.
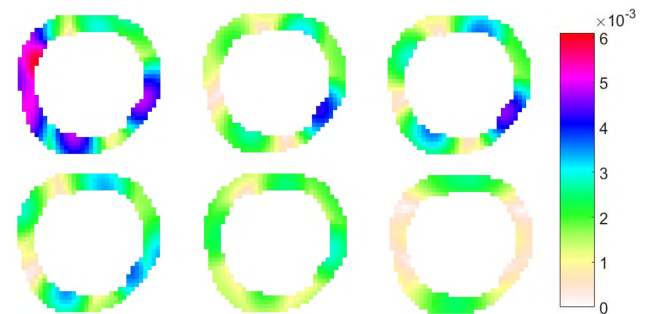
We also compared the results obtained using MIDGM and state-of-the-art methods on the HSC dataset. References [55] and [57] also used this dataset to validate their methods. Wilms *et al.* [55] extracted midventricular slices at the ED phase and trained them using patch-based active appearance models. However, only the ED slices are segmented to delineate the LV's endocardium and epicardium. The SCD is 1.62 *mm*, whereas that of our method is 1.37 *mm*. Uzunova *et al.* [57] adapted the FlowNet architecture for CNN-based optical flow estimation to register slices at the ED phase. Their algorithm's performance was evaluated by computing the average SCDs for the right ventricle (RV) and LV endocardium and LV epicardium contours. Ehrhardt *et al.* [58] conducted experiments showing that the mean SCD of the RV is approximately 3 *mm* and the mean SCD of LV is approximately 2 *mm* for the HSC dataset.
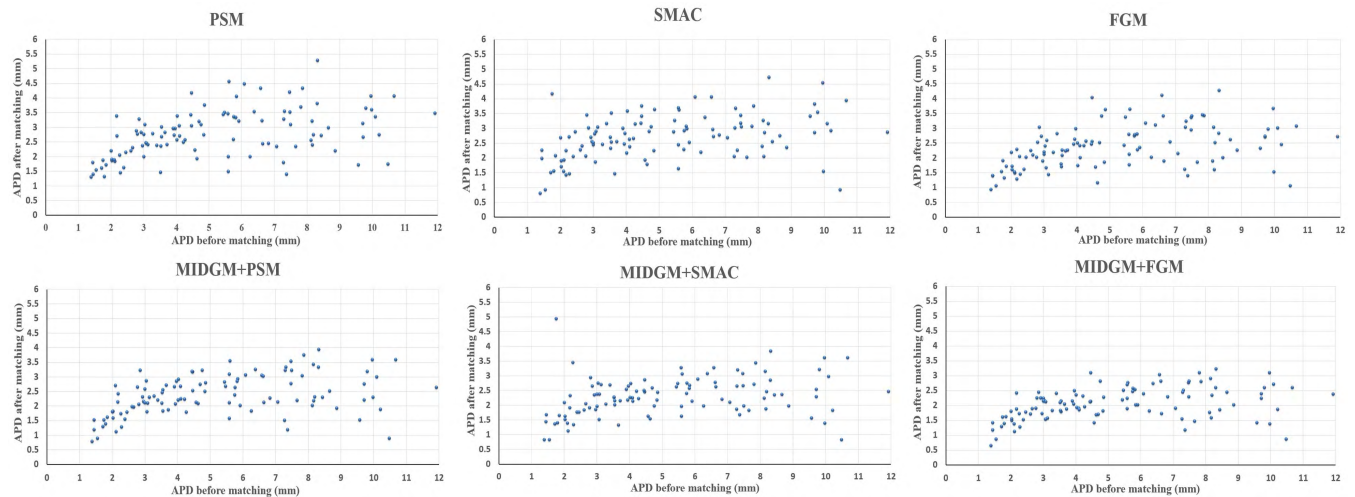
FIGURE 10. APDs before and after matching. Each point corresponds to an APD of the LV contours between ES and ED. The *x*-axis indicates APDs before matching and the *y*-axis indicates APDs after matching. A dense set of points with a smaller value along the *y*-axis indicate improved APDs. The first row shows the APDs before and after matching using PSM, SMAC and FGM, and the second row shows the results using MIDGM + PSM, MIDGM + SMAC and MIDGM + FGM.

Based on this conclusion, we determined the mean SCD of the LV to be approximately 2.29 *mm* in [57]. The same experiment was performed using [27] to compare the performance of our method with the point set matching algorithm. Reference [27] is a very effective point set registration method and its performance has been judged to be very good for several publicly available datasets. The mean SCD and ACC between the target and mapped source contours were used to evaluate the algorithms' performance. Table 4 lists the SCD and the ACC of the algorithms from [27], [55], and [57] and our MIDGM + FGM method, respectively. Note that [27] outperforms [55], [57] and our MIDGM + FGM with respect to SCD, but with low correspondence accuracy. Here, we emphasize that the correspondence between points is important in motion estimation because the deformation field is controlled by this correspondence, implying that the point matching algorithm is not applicable to LV motion estimation, because only the point position is used in point matching algorithms, not the objects' structural information. This result validates the effectiveness of the GM algorithms in LV motion estimation. The results in Table 4 show that our method achieves a good balance between registration accuracy and correspondence.

2) MICCAI2009 CHALLENGE DATASET

To further verify the performance of our MIDGM algorithm, cine MRI cardiac data provided by Sunnybrook Health Sciences Centre [54] were employed to evaluate the performance of LV motion estimation. Using our MIDGM model, PSM, SMAC, and FGM, we registered slices at the ES phase to their corresponding slices at the ED phase, and then estimated the correspondence and spatial transformations between these point sets.

TABLE 4. Comparison of mean symmetric contour distances and correspondence accuracy using our method and state-of-the-art algorithms for the HSC dataset, with optimal results in bold. (mean_value ± standard_deviation, A hyphen ("−") denotes no operation is performed.)

Evaluation	Methods			
	[57]	[55]	MIDGM + FGM	[27]
SCD	2.67±0.93	1.62±0.70	1.37±0.52	**1.22±0.43**
ACC	—	—	**0.68**	0.31

Table 5 lists the mean values of the APD, BE, and SST for 30 cases from the MICCAI2009 dataset using different algorithms. Note that the APD values using MIDGM are significantly better than those of other GM algorithms, indicating that MIDGM is more effective at estimating LV motion than traditional GM algorithms, insofar as the APD is one of the key metrics in LV motion estimation. Moreover, deformation fields were smoother using MIDGM because of its lower BE and SST than other algorithms. These evaluation indicators show that our proposed MIDGM method is more effective in LV motion estimation than state-of-the-art algorithms.

The visualized APDs of all LV contours before and after matching are shown in Fig. 10. Here, the APDs of all LV contours for 30 cases are provided. The first row shows the APDs before and after matching using PSM, SMAC, and FGM, where the *x*-axis indicates APDs before matching, and the *y*-axis indicates APDs after matching. The second row shows the corresponding results using MIDGM + PSM, MIDGM + SMAC, and MIDGM + FGM, respectively. Fig. 10 shows the improvement in the APD with our MIDGM. The graphs show a clear improvement for large deformations (i.e., those with more than 5 mm prior to matching). For the PSM, for example, most APDs are below 3 along the *y*-axis using

TABLE 5. Performance of LV motion estimation, comparing our MIDGM to three state-of-the-art graph matching algorithms using the Sunnybrook dataset, with optimal results in bold. Note: SST (*value* $\times 10^{-4}$), APD (*mm*).

Cases	Evaluation	Methods					
		PSM	MIDGM + PSM	SMAC	MIDGM + SMAC	FGM	MIDGM + FGM
NUM^1	APD	2.84	**2.43**	2.79	**2.27**	2.47	**2.18**
	BE	1.43	**0.79**	1.16	**0.65**	0.92	**0.48**
	SST	1.83	**1.59**	1.40	**1.17**	1.51	**1.39**
NUM^2	APD	2.32	**2.11**	2.82	**2.08**	2.16	**1.94**
	BE	1.69	**1.24**	1.47	**1.12**	1.23	**0.97**
	SST	1.84	**0.56**	1.89	**0.61**	1.25	**0.81**
NUM^3	APD	2.58	**2.27**	2.81	**2.18**	2.32	**2.06**
	BE	1.56	**1.02**	1.32	**0.89**	1.08	**0.61**
	SST	1.84	**1.08**	1.65	**0.89**	1.38	**1.10**

NUM^1: Number of datasets evaluated. 15–test datasets. NUM^2: 15–online datasets. NUM^3: 30–test and online datasets.

TABLE 6. The APD of LV segmentation of the MICCAI 2009 challenge dataset using different methods, with optimal results in bold.

Evaluation	Methods			
	[59]	[60]	[61]	MIDGM + FGM
NUM^1	45	45	15	45
APD	2.19	2.14	2.23	**1.98**

NUM^1: Number of datasets evaluated. 15–test datasets, 30–test and online datasets, 45–all datasets.

MIDGM + PSM, whereas they are predominantly below 3.5 using PSM. This indicates that the LV motion accuracy using MIDGM + PSM is obviously better than it is using PSM. The second and third columns present similar results, but in different ways. Note that our MIDGM method achieves more accurate results.

In addition, three state-of-the-art algorithms, [59]–[61], were compared with our method on the MICCAI 2009 challenge dataset. Table 6 summarizes the APD of the experimental results obtained using our method and the other three algorithms on the MICCAI2009 challenge dataset. Note that [59] and [60], and our method performed on all 45 cases, whereas method [61] was only applied to 15 cases. However, our method still achieved more accurate results than the other state-of-the-art algorithms.

VII. CONCLUSION

Early detection of abnormalities in cardiac motion is important for improved medical treatment. In this paper, we have described a novel multilayer iterative deformable GM algorithm for LV motion estimation. To make GM algorithms suitable for applications with large deformations, we introduced a transformation model regularized by l_1 and l_2 norms to the GM algorithms. With our proposal, spatial transformation is less sensitive to mismatched nodes, a smooth deformation field is maintained, and the topology is preserved. Moreover, by alternately estimating the correspondence between nodes and the spatial transformation, GM can produce an accurate correspondence. We also proposed a multilayer framework

to improve the correspondence accuracy. An evaluation performed using the HSC dataset and the MICCAI2009 Challenge for Clinical Applications dataset demonstrated that our multilayer iterative deformable GM method is effective in estimating LV motion. The method we developed can be applied to clinically obtained data and is applicable in a clinical environment. As the point set is extracted from the contours of the LV, the accuracy of motion estimation is dependent on the segmentation of the LV myocardium. To improve segmentation accuracy, a modern deep learning network will be needed. Furthermore, more sophisticated features for describing nodes and edges are needed to measure the similarity between points and between edges of the LV myocardium.

REFERENCES

[1] M. Alessandrini *et al.*, "A new technique for the estimation of cardiac motion in echocardiography based on transverse oscillations: A preliminary evaluation in silico and a feasibility demonstration *in vivo*," *IEEE Trans. Med. Imag.*, vol. 33, no. 5, pp. 1148–1162, May 2014.

[2] M. G. Danilouchkine, J. J. M. Westenberg, A. de Roos, J. H. C. Reiber, and B. P. F. Lelieveldt, "Operator induced variability in cardiovascular MR: Left ventricular measurements and their reproducibility," *AIP Adv.*, vol. 7, no. 2, pp. 447–455, 2015.

[3] M. Tee, J. A. Noble, and D. A. Bluemke, "Imaging techniques for cardiac strain and deformation: Comparison of echocardiography, cardiac magnetic resonance and cardiac computed tomography," *Expert Rev. Cardiovascular Therapy*, vol. 11, no. 2, pp. 221–231, Feb. 2013.

[4] R. Dharmakumar, J. M. Arumana, R. Tang, K. Harris, Z. Zhang, and D. Li, "Assessment of regional myocardial oxygenation changes in the presence of coronary artery stenosis with balanced SSFP imaging at 3.0 T: Theory and experimental evaluation in canines," *J. Magn. Reson. Imag.*, vol. 27, no. 5, pp. 1037–1045, 2008.

[5] M. J. Ledesma-Carbayo *et al.*, "Spatio-temporal nonrigid registration for ultrasound cardiac motion estimation," *IEEE Trans. Med. Imag.*, vol. 24, no. 9, pp. 1113–1126, Sep. 2005.

[6] F. M. Parages, T. S. Denney, H. Gupta, S. G. Lloyd, L. J. Dell'italia, and J. G. Brankov, "Estimation of left ventricular motion from cardiac gated tagged MRI using an image-matching deformable mesh model," *IEEE Trans. Radiation Plasma Med. Sci.*, vol. 1, no. 2, pp. 147–157, Mar. 2017.

[7] P. Clarysse and P. Croisille, "Cardiac motion analysis in tagged MRI," in *Multi-Modality Cardiac Imaging*. Hoboken, NJ, USA: Wiley, 2015, pp. 247–255.

[8] W. Shi *et al.*, "A comprehensive cardiac motion estimation framework using both untagged and 3-D tagged MR images based on nonrigid registration," *IEEE Trans. Med. Imag.*, vol. 31, no. 6, pp. 1263–1275, Jun. 2012.

[9] R. Chandrashekara, R. Mohiaddin, and D. Rueckert, "Comparison of cardiac motion fields from tagged and untagged MR images using non-rigid registration," in *Proc. FIMH*. Barcelona, Spain: Springer, 2005, pp. 425–433.

[10] D. Perperidis, R. H. Mohiaddin, and D. Rueckert, "Spatio-temporal free-form registration of cardiac MR image sequences," *Med. Image Anal.*, vol. 9, no. 5, pp. 441–456, 2005.

[11] H. Sundar, H. Litt, and D. Shen, "Estimating myocardial motion by 4D image warping," *Pattern Recognit.*, vol. 42, no. 11, pp. 2514–2526, 2009.

[12] N. S. Phatak, S. A. Maas, A. I. Veress, N. A. Pack, E. V. R. Di Bella, and J. A. Weiss, "Strain measurement in the left ventricle during systole with deformable image registration," *Med. Image Anal.*, vol. 13, pp. 354–361, Sep. 2009.

[13] K. McLeod, M. Sermesant, P. Beerbaum, and X. Pennec, "Spatio-temporal tensor decomposition of a polyaffine motion model for a better analysis of pathological left ventricular dynamics," *IEEE Trans. Med. Imag.*, vol. 34, no. 7, pp. 1562–1575, Jul. 2015.

[14] M. Ebrahimi and S. Kulaseharan, "Deformable image registration and intensity correction of cardiac perfusion MRI," in *Proc. Int. Workshop Stat. Atlases Comput. Models Heart*. Springer, 2015, pp. 13–20.

[15] B. Gao *et al.*, "Estimation of cardiac motion in cine-MRI sequences by correlation transform optical flow of monogenic features distance," *Phys. Med. Biol.*, vol. 61, no. 24, p. 8640, 2016.

[16] M. M. Lamacie *et al.*, "Quantification of global myocardial function by cine MRI deformable registration-based analysis: Comparison with MR feature tracking and speckle-tracking echocardiography," *Eur. Radiol.*, vol. 27, no. 4, pp. 1404–1415, 2017.

[17] N. Lin, X. Papademetris, A. J. Sinusas, and J. S. Duncan, "Analysis of left ventricular motion using a general robust point matching algorithm," in *Proc. MICCAI*. Montreal, QC, Canada: Springer, 2003, pp. 556–563.

[18] N. Lin and J. S. Duncan, "Generalized robust point matching using an extended free-form deformation model: Application to cardiac images," in *Proc. 2nd IEEE Int. Symp. Biomed. Imag., Nano Macro*, Apr. 2004, pp. 320–323.

[19] P. Yan, A. Sinusas, and J. S. Duncan, "Boundary element method-based regularization for recovering of LV deformation," *Med. Image Anal.*, vol. 11, no. 6, pp. 540–554, 2007.

[20] Y. Zhu, X. Papademetris, A. J. Sinusas, and J. S. Duncan, "Integrated segmentation and motion analysis of cardiac MR images using a subject-specific dynamical model," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2008, pp. 1–8.

[21] E. W. Remme, K. F. Augenstein, A. A. Young, and P. J. Hunter, "Parameter distribution models for estimation of population based left ventricular deformation using sparse fiducial markers," *IEEE Trans. Med. Imag.*, vol. 24, no. 3, pp. 381–388, Mar. 2005.

[22] S. Randrianarisolo, E. Delechelle, E. Petit, A. Rahmouni, and J. Garot, "Assessment of myocardial deformations from untagged cardiac cine MRI," in *Proc. 30th Annu. Int. Conf. Eng. Med. Biol. Soc. (EMBS)*, Aug. 2008, pp. 3401–3404.

[23] H. Wang and A. A. Amini, "Cardiac motion and deformation recovery from MRI: A review," *IEEE Trans. Med. Imag.*, vol. 31, no. 2, pp. 487–503, Feb. 2012.

[24] A. Bistoquet, J. Oshinski, and O. Skrinjar, "Left ventricular deformation recovery from cine MRI using an incompressible model," *IEEE Trans. Med. Imag.*, vol. 26, no. 9, pp. 1136–1153, Sep. 2007.

[25] T. Mansi, X. Pennec, M. Sermesant, H. Delingette, and N. Ayache, "iLogDemons: A demons-based registration algorithm for tracking incompressible elastic biological tissues," *Int. J. Comput. Vis.*, vol. 92, no. 1, pp. 92–111, 2011.

[26] J. Schaerer, C. Casta, J. Pousin, and P. Clarysse, "A dynamic elastic model for segmentation and tracking of the heart in MR image sequences," *Med. Image Anal.*, vol. 14, no. 6, pp. 738–749, 2010.

[27] J. Ma, J. Wu, J. Zhao, J. Jiang, H. Zhou, and Q. Z. Sheng, "Nonrigid point set registration with robust transformation learning under manifold regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published.

[28] J. Ma, J. Zhao, J. Jiang, H. Zhou, and X. Guo, "Locality preserving matching," *Int. J. Comput. Vis.*, vol. 127, no. 5, pp. 512–531, 2019.

[29] J. Ma, J. Zhao, and A. L. Yuille, "Non-rigid point set registration by preserving global and local structures," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 53–64, Jan. 2016.

[30] J. Ma, W. Qiu, J. Zhao, Y. Ma, A. L. Yuille, and Z. Tu, "Robust L2E estimation of transformation for non-rigid registration," *IEEE Trans. Signal Process.*, vol. 63, no. 5, pp. 1115–1129, Mar. 2015.

[31] T. Cour, P. Srinivasan, and J. Shi, "Balanced graph matching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 313–320.

[32] F. Zhou and F. De La Torre, "Factorized graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1774–1789, Sep. 2016.

[33] G. L. Scott and H. C. Longuet-Higgins, "An algorithm for associating the features of two images," *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 244, no. 1309, pp. 21–26, 1991.

[34] L. S. Shapiro and J. M. Brady, "Feature-based correspondence: An eigenvector approach," *Image Vis. Comput.*, vol. 10, no. 5, pp. 283–288, 1992.

[35] S. Umeyama, "An eigendecomposition approach to weighted graph matching problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 5, pp. 695–703, Sep. 1988.

[36] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 2, Oct. 2005, pp. 1482–1489.

[37] M. Leordeanu, M. Hebert, and R. Sukthankar, "An integer projected fixed point method for graph matching and MAP inference," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2009, pp. 1114–1122.

[38] C. Leng, W. Xu, I. Cheng, and A. Basu, "Graph matching based on stochastic perturbation," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4862–4875, Dec. 2015.

[39] C. Schellewald and C. Schnörr, "Probabilistic subgraph matching based on convex relaxation," in *Proc. Int. Workshop Energy Minimization Methods Comput. Vis. Pattern Recognit.* San Diego, CA, USA: Springer, 2005, pp. 171–186.

[40] X. Bai, H. Yu, and E. R. Hancock, "Graph matching using spectral embedding and semidefinite programming," in *Proc. BMVC*, 2004, pp. 1–10.

[41] M. Zaslavskiy, F. Bach, and J. P. Vert, "A path following algorithm for the graph matching problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2227–2242, Dec. 2009.

[42] Z.-Y. Liu, H. Qiao, and L. Xu, "An extended path following algorithm for graph-matching problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1451–1456, Jul. 2012.

[43] B. Jiang, H. Zhao, J. Tang, and B. Luo, "A sparse nonnegative matrix factorization technique for graph matching problems," *Pattern Recognit.*, vol. 47, no. 2, pp. 736–747, 2014.

[44] A. C. Berg, T. L. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondences," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 26–33.

[45] F. Zhou and F. De la Torre, "Deformable graph matching," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2922–2929.

[46] X. Yang, B. Wang, Y.-R. Li, and T. He, "Robust landmark-based image registration using l1 and l2 norm regularizations," in *Proc. IEEE Int. Conf. Bioinformatics Biomed.*, Nov. 2015, pp. 425–428.

[47] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[48] P. J. Besl and D. N. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[49] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Comput. Vis. Image Understand.*, vol. 89, nos. 2–3, pp. 114–141, Feb. 2003.

[50] H. Wendland, "On the smoothness of positive definite and radial functions," *J. Comput. Appl. Math.*, vol. 101, nos. 1–2, pp. 177–188, 1999.

[51] Z. Li, A. Lin, X. Yang, and J. Wu, "Left ventricle segmentation by combining convolution neural network with active contour model and tensor voting in short-axis MRI," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, Nov. 2017, pp. 736–739.

[52] W. Guo, X. Yang, J. Wu, and A. Lin, "Left ventricle motion estimation for cardiac cine MRI using graph matching," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, Nov. 2017, pp. 724–727.

[53] A. Andreopoulos and J. K. Tsotsos, "Efficient and generalizable statistical models of shape and appearance for analysis of cardiac MRI," *Med. Image Anal.*, vol. 12, no. 3, pp. 335–357, 2008.

[54] P. Radau, Y. Lu, K. Connelly, G. Paul, A. Dick, and G. Wright, "Evaluation framework for algorithms segmenting short axis cardiac MRI," in *Proc. MIDAS J. Cardiac MR Left Ventricle Segmentation Challenge*, vol. 49, 2009.

[55] M. Wilms, H. Handels, and J. Ehrhardt, "Representative patch-based active appearance models generated from small training populations," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2017, pp. 152–160.

[56] J. Yi, H. Yang, X. Yang, and G. Chen, "Lung motion estimation by robust point matching and spatiotemporal tracking for 4D CT," *Comput. Biol. Med.*, vol. 78, pp. 107–119, Nov. 2016.

[57] H. Uzunova, M. Wilms, H. Handels, and J. Ehrhardt, "Training CNNs for image registration from few samples with model-based data augmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Springer, 2017, pp. 223–231.

[58] J. Ehrhardt, T. Kepp, A. Schmidt-Richberg, and H. Handels, "Joint multi-object registration and segmentation of left and right cardiac ventricles in 4D cine MRI," *Proc. SPIE*, vol. 9034, Mar. 2014, Art. no. 90340M.

[59] M. Irshad, N. Muhammad, M. Sharif, and M. Yasmeen, "Automatic segmentation of the left ventricle in a cardiac mr short axis image using blind morphological operation," *Eur. Phys. J. Plus*, vol. 133, no. 4, p. 148, 2018.

[60] S. Queirós *et al.*, "Fast automatic myocardial segmentation in 4D cine CMR datasets," *Med. Image Anal.*, vol. 18, no. 7, pp. 1115–1131, 2014.

[61] X. Yang, Q. Song, and S. Yi, "Automatic segmentation of left ventricle cavity from short-axis cardiac magnetic resonance images," *Med. Biol. Eng. Comput.*, vol. 55, no. 9, pp. 1563–1577, 2017.

JUNHAO WU received the bachelor' degree from the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, Guangdong, China, in 2015, where he is currently pursuing the Ph.D. degree. His research interests include image registration, parallel computing, and deep learning.
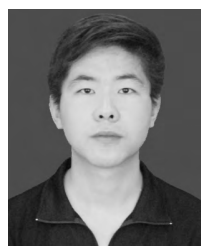
ZHENGRUI ZHANG received the M.S. degree from the College of Information Engineering, Shenzhen University, Shenzhen, Guangdong, China, in 2014, where he is currently pursuing the Ph.D. degree with the College of Information Engineering. His research interests include image registration, image segmentation, and deep learning.

XUAN YANG received the Ph.D. degree in communication and information system from Xi'an Jiaotong University, Xi'an, China, in 1998. She was with the Postdoctoral Mobile Station of Electronic Science and Technology, Xi'an University, from 1999 to 2001.

She has been involved in teaching and scientific research at Shenzhen University, since 2001. Her current research interests include image processing, medical image processing, and deep learning.

GUOLIANG CHEN received the B.Sc. degree from Xi'an Jiaotong University, Xi'an, China, in 1961.

Since 1973, he has been with the University of Science and Technology of China, Hefei, China, where he is currently the Academic Committee Chair of the Nature Inspired Computation and Applications Laboratory, a Professor with the Department of Computer Science and Technology, and the Director of the School of Software Engineering. From 1981 to 1983, he was a Visiting Scholar with Purdue University, West Lafayette, IN, USA. He is also the Director of the National High Performance Computing Center at Hefei, Hefei. He is currently the Director of the College of Computer Science and Software Engineering, Shenzhen University. He has published nine books and more than 200 research papers. His research interests include parallel algorithm, computer architecture, computer networks, and computational intelligence.

Prof. Chen is an Academician of the Chinese Academy of Sciences. He was a recipient of the National Excellent Teaching Award of China, in 2003.

• • •