A Boundary Element-Based Approach
to Analysis of LV Deformation

Ping Yan[1,3], Ning Lin[1,3], Albert J. Sinusas[2,4], and James S. Duncan[1,2,3]

Departments of [1] Electrical Engineering,
[2] Diagnostic Radiology,
[3] Biomedical Engineering and [4] Medicine,
Yale University PO Box 208042, New Haven, CT 06520-8042, USA
ping.yan@yale.edu

Abstract. Quantification of left ventricular (LV) deformation from 3D image sequences (4D data) is important for the assessment of myocardial viability, which can have important clinical implications. To date, feature information from either Magnetic Resonance, computed tomographic or echocardiographic image data has been assembled with the help of different interpolative models to estimate LV deformation. These models typically are designed to be computationally efficient (e.g. regularizing strategies using B-splines) or more physically realistic (e.g. finite element approximations to biomechanical models), but rarely incorporate both notions. In this paper, we combine an approach to the extraction and matching of image-derived point features based on local shape properties with a boundary element model. This overall scheme is intended to be both computationally efficient and physically realistic. In order to illustrate this, we compute strains using our method on canine 4D MR image sequences and compare the results to those found from a B-spline-based method (termed extended free-form deformation (EFFD)) and a method based on finite elements (FEM). All results are compared to displacements found using implanted markers, taken to be a gold standard.

1 Introduction

Quantitative analysis of left ventricular (LV) deformation is known to be a sensitive index of myocardial ischemia and injury. However, while there have been many methods proposed for the estimation of LV deformation [1, 2, 6, 7], most employ some form of modeling to interpolate dense displacement fields from sparse image derived features. These features include shape-based measures [1], MR tags [3], MR phase velocity [4] and echocardiographic features [5]. These models suffer from an inherent tradeoff between the computation time and the complexity of the model. A biomechanical model constructed using FEM, which incorporates the microstructure of the LV, is considered as the model closest to the physical reality of the LV but solving for the parameters embedded in these models is usually time-consuming [2,6]. Recently, simple deformation models such as B-spline-based EFFD [8] have been developed to estimate the LV deformation [7]. This method is computationally efficient but typically doesn't reflect the true physical properties of LV. Hence these

models are typically less accurate. This line of reasoning leads us to the following questions: Is it reasonable to trade off the approximation of physical reality with the computation time? Is there a method that is computationally more efficient than FEM yet still is used to directly incorporate physical parameters?

Over the several past decades, the Boundary Element Method (BEM) has emerged as a versatile and powerful alternative to FEM for the solution of engineering problems [9]. The most important advantage of BEM over the FEM is that it only requires discretization of the surface rather than the volume. Hence it can speed up the solution and it is easier to generate a mesh. The partitioning of only the boundary also makes BEM very suited to applications in the LV motion estimation problem where most feature points are on the boundary. Thus it is our view in this paper that BEM comes close to the physical reality of FEM modeling but is closer to EFFD in computation time.

Our approach, as described in this paper, combines an approach to shape-based feature tracking termed generalized robust point matching (G-RPM) [10] with a biomechanical model constructed using BEM to derive the dense strains from LV image sequences. The attractiveness of G-RPM is its ability to estimate non-rigid correspondences with only one-time *rough* segmentation in the first frame (see [10]).

To our knowledge no prior work has been reported that compares a B-spline-based deformation model with the biomechanical model in this domain. In this paper, we present the comparisons among the B-spline-based EFFD, and biomechanical model-based BEM and FEM.

The organization of the paper is as follows. In Section 2, we first briefly review the G-RPM algorithm and BEM. The computational cost of FEM and BEM are then compared in this section. In Section 3, Canine MRI cardiac datasets are analyzed using our algorithm. We estimate each algorithm's accuracy by comparing the results to the displacements found using implanted markers. Meanwhile, the comparisons of strain results among the FEM, BEM and EFFD are presented in this section. Finally, Section 4 discusses the results and proposed future research.

2   Methods

2.1   The G-RPM Algorithm

G-RPM is an extension of robust point matching (RPM) [7]. We have previously reported results using this approach with B-spline in [7]. In this work, we embedded shape-based information into RPM and achieved more accurate results than when using point distances alone. Assume we have a data point-set $X = \{x_i, i = 1, 2, \ldots, N\}$ and a template point-set $Y = \{y_a, a = 1, 2, \ldots, K\}$. We denote the outlier cluster as x_{N+1} and y_{K+1}, and their unique temperature as T_0. To match set Y onto X, it proposed to minimize the following objective energy function:

$$E(M, \alpha) = \sum_{i=1}^{N+1} \sum_{a=1}^{K+1} m_{ai} \left[\left\| x_i - f(y_a, \alpha) \right\|^2 + \lambda_A g \left(\left\| A_X(x_i) - A_Y(y_a) \right\|^2 \right) \right] + \lambda \left\| Lf \right\|^2 +$$

$$\zeta \sum_{i=1}^{N} \sum_{a=1}^{K} m_{ai} + T \sum_{i=1}^{N} \sum_{a=1}^{K} m_{ai} \log m_{ai} + T_0 \sum_{a=1}^{K} m_{a,N+1} \log m_{a,N+1} + T_0 \sum_{i=1}^{N} m_{K+1,i} \log m_{K+1,i}$$

(1)

Where f is a non-rigid transformation function with parameters α. T is the annealing temperature, gradually decreasing to zero as the matching iteration begins. Correspondence matrix M or $m_{ai} \in [0,1]$ is subject to the constraints: $\sum_{i=1}^{N+1} m_{ai} = 1$, for $a \in \{1,2,...,K\}$ and $\sum_{a=1}^{K+1} m_{ai} = 1$, for $i \in \{1,2,...,N\}$. $g(\cdot)$ is a strictly increasing function and λ_A balances the significance between the distance and new information. $A_X(x_i)$ and $A_Y(y_a)$ are the curvatures of two point-sets, respectively.

Minimization of equation 1 is solved by an alternating update process where two steps are repeated until the algorithm converges. One step is to update the correspondence matrix m_{ai}. The closed form solution can be obtained by differentiating the objective function (1) w.r.t m_{ai} and setting the result to zero:

$$m_{ai} = e^{-\frac{\left[\|x_i - f(y_a,\alpha)\|^2 + \lambda_A g(\|A_X(x_i) - A_Y(y_a)\|^2)\right] - \zeta}{T}} \tag{2}$$

A second step to update transformation parameter α involves a least-squared approach given the m_{ai}:

$$\min_{\alpha} E(\alpha) = \min_{\alpha} \sum_{\alpha=1}^{K} \frac{\|\hat{x}_a - f(y_a,\alpha)\|^2}{\sigma_a^2} + \lambda \|Lf\|^2 \tag{3}$$

Where $\hat{x}_a = \sum_{i=1}^{N} m_{ai} x_i / \sum_{i=1}^{N} m_{ai}$ and $1/\sigma_a^2 = \sum_{i=1}^{N} m_{ai}$. In this algorithm, we used curvature derived from area-based operators [10]. For example, the Gaussian curvature κ from 3D volume data is given by:

$$\kappa = \frac{L_x^2 L_{yy} L_{zz} + L_y^2 L_{xx} L_{zz} + L_z^2 L_{xx} L_{yy}}{(L_x^2 + L_y^2 + L_z^2)^2} \tag{4}$$

Where $L(x, y, z)$ denotes image function. L_x, L_y and L_z are the first derivatives of L in x, y and z, and L_{xx}, L_{yy} and L_{zz} are the second partials in x, y and z.

2.2 The Boundary Element Method

The boundary element method is a technique for solving partial differential equations by reformulating the original PDE into an integral equation over the boundary of a solid object [9]. Because the integral equations are over the boundary of the object, only the boundary of the object needs to be partitioned. The elastic problem can be expressed by Navier's equation:

$$\mu \nabla^2 \vec{u} + \frac{\mu}{1 - 2\nu} \nabla(\nabla \cdot \vec{u}) = 0 \tag{5}$$

Where μ and ν are the shear modulus and Poisson's ratio of the elastic material. $\vec{u}$ is the displacement vector. The formulation of the boundary integral equations for elastic problems requires the knowledge of the solution of the elastic problems with the same material properties as the body under consideration but corresponding to an infinite domain loaded with a concentrated unit point load. This is called the fundamental solution. The integral equation that relates interior displacements to boundary displacements and tractions (surface force) is known as Somigliana's identity [9]:

$$u_l^i + \int_\Gamma p_{lk}^* u_k d\Gamma = \int_\Gamma u_{lk}^* p_k d\Gamma \tag{6}$$

Where u_{lk}^* and p_{lk}^* represent the fundamental solutions of displacement and traction at any point in the k direction when a unit load is applied at 'i' in the l direction. $\vec{p}$ is traction vector. Γ is the boundary. When 'i' is taken to the boundary, equation 6 transforms into:

$$c_{lk}^i u_l^i + \int_\Gamma p_{lk}^* u_k d\Gamma = \int_\Gamma u_{lk}^* p_k d\Gamma \tag{7}$$

When 'i' is at a point where the boundary is smooth, $c_{lk}^i = \dfrac{1}{2}\delta_{lk}$. $\delta_{lk} = \begin{cases} 1, l = k \\ 0, l \neq k \end{cases}$

In order to solve the integral equation numerically, the boundary will be discretized into a series of elements. We can start by defining the $\vec{u}$ and $\vec{p}$ which apply over each element 'j', i.e. $\vec{u} = \Phi\vec{u}^j$ and $\vec{p} = \Phi\vec{p}^j$. Φ is the interpolation function matrix of shape functions. To study curved elements first we need to define the way in which we can pass from the global Cartesian system {x, y, z} to the local system {ξ₁, ξ₂, η} defined over the element, where ξ_1, ξ_2 are oblique coordinates and η is in the direction of the normal. The Jacobian of transformation is $d\Gamma = |J| d\xi_1 d\xi_2$. Equation 7 can now be written as:

$$c_{lk}^i \vec{u}^i + \sum_{j=1}^N \left\{ \int_{\Gamma_j} p_{lk}^* \Phi |J| d\xi_1 d\xi_2 \right\} \vec{u}^j = \sum_{j=1}^N \left\{ \int_{\Gamma_j} u_{lk}^* \Phi |J| d\xi_1 d\xi_2 \right\} \vec{p}^j \tag{8}$$

It can also be expressed in matrix form as follows:

$$\sum_{j=1}^N H^{ij} \vec{u}^j = \sum_{j=1}^N G^{ij} \vec{p}^j \tag{9}$$

Where $H^{ij} = \hat{H}^{ij}$, if i ≠ j and $H^{ij} = \hat{H}^{ij} + c^i$, if i=j.

Somigliana's identity gives the displacement at any internal point in terms of the boundary displacements $\vec{u}^j$ and tractions $\vec{p}^j$ of each element:

$$\vec{u}^i = \sum_{j=1}^N G^{ij} \vec{p}^j - \sum_{j=1}^N H^{ij} \vec{u}^j \tag{10}$$

2.3 Comparison of the Operation Counts

In our algorithm, the H and G matrix only need to be calculated once in analyzing image sequences. Then displacements of any internal point at any frame can be computed by the equation 9 and 10 which don't need much computation time. For a L-sequence problem with N boundary nodes and M internal nodes, the operation counts of BEM and FEM are $O(N^3)+O(M^3)+O(L(NM))$ and $O(L(N+M)^3)$, respectively. Therefore, the BEM is computationally more efficient than the FEM, especially when we analyze image sequences.

2.4 Implementation

Feature Point Extraction: Not all points in images are suitable candidates for tracking. We choose our feature points as follows: 1) to stabilize curvature values, discard the points whose edge strength $\|\nabla L\|$ is less than a threshold; 2) thin the wide ridges resulting from the previous step by nonmaximum suppression [12] and 3) choose the resulting local maxima as the final candidate points to track.

Fundamental Solutions for Elastic Problem: In isotropic elastic materials, the fundamental solutions are known in exact closed form as shown in equation 11 and 12 [9]. In anisotropic analysis, the closed form of a fundamental solution is hard to get and calculation becomes very time-consuming. However, by using the Wilson & Cruse interpolation scheme, computation time is almost the same. (See [13]). For the 3D isotropic elastic problem, the fundamental solutions of displacement u_{lk}^* and traction p_{lk}^* are:

$$u_{lk}^* = \frac{1}{16\pi\mu(1-\nu)\,|\,r\,|}[(3-4\nu)\delta_{lk}+r_{,l}r_{,k}] \tag{11}$$

$$p_{lk}^* = \frac{1}{8\pi(1-\nu)\,|\,r\,|^2}\left[\frac{\partial r}{\partial n}\Big[(1-2\nu)\delta_{lk}+3r_{,l}r_{,k}\Big]+(1-2\nu)(n_l r_{,k}-n_k r_{,l})\right] \tag{12}$$

Where r is the distance between the loading point to any point under consideration. Notice that $r_{,l}$ is the derivative of the r in the l direction. n_l and n_k are the direction cosines of the normal with respect to l and k direction. In this paper, we used $\mu = 17857.1$ and $\nu = 0.4$ for the LV myocardium.

Strain tensor definition: Given a strain tensor E (a 3×3 matrix) which is computed in the Cartesian coordinate system of {x, y, z}. We can transform it to a local coordinate system $\{\xi_1, \xi_2, \eta\}$. First construct the 3×3 rotation matrix R which results in {x, y, z} → $\{\xi_1, \xi_2, \eta\}$. Using this matrix R we can write the E in the local coordinate system as: $E_{local} = RER'$. In this paper, the local coordinate system is defined by the circumferential, radial, and longitudinal axis at the epicardium.
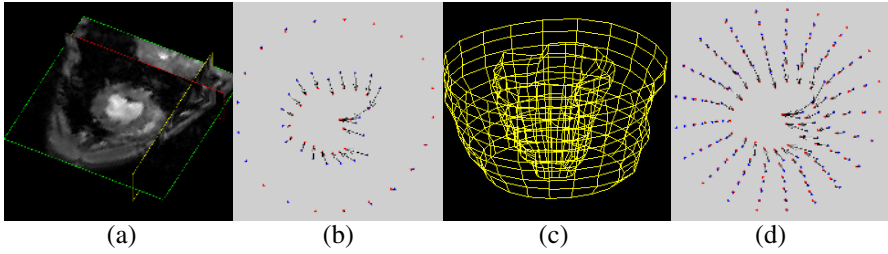
Fig. 1. (a) Original 3D MRI Data in the Orthogonal-slice view, (b) Boundary displacement results for one of its 2D slices obtained using G-RPM, (c) 3D BEM mesh using bilinear element, (d) Dense displacement results for one 2D slices through the 3D results.

3   Experimental Results

4D MR canine cardiac datasets were analyzed using our new algorithm. An example of the results from one of our datasets was shown in figure 1. For each dataset, strains were calculated between end-diastole (ED) and end-systole (ES). The resulting 3D-derived radial strains (E_{rr}) and Circumferential strains (E_{cc}) of one dataset at frame 1, 3, 4 and 6 were illustrated in figure 2. Frame 1 was at ED and Frame 6 was at ES. Note the normal behavior in the LV, showing positive radial strain (thickening) and negative circumferential strain (shortening) as we move from ED to ES.
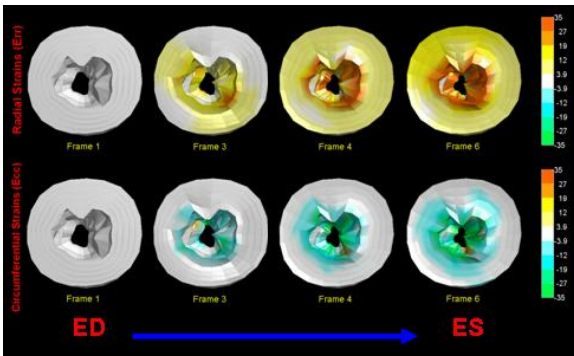

Fig. 2. E_{rr} (Top) and E_{cc} (Bottom) between ED (frame 1) to ES (frame 6) of dataset 1

3.1   Comparisons with Implanted Markers

To further quantitatively validate resulted motion trajectories, we used four canine MRI datasets with implanted markers for point-by-point displacement comparison (see [6] for more details on the marker implantation and localization.). The mean displacement errors of FEM, BEM and EFFD methods were illustrated in the figure 3. It can be seen that FEM-based models show the smallest error while the EFFD approach shows the largest.

3.2   Comparisons Between FEM and BEM, FEM and EFFD

In order to see the difference between the B-Spline based deformation model and the biomechanical model, we compared strain results obtained using FEM, BEM and

EFFD. The strains were calculated between ED to ES using BEM, FEM and EFFD, respectively. The comparisons of strain results between FEM and BEM, FEM and EFFD for 4 canine image sequences are illustrated in the figure 3. The mean differences of E_{cc}, E_{rr} and E_{ll} between FEM and BEM were less than 1.3%, 4.05% and 4.24%, while the mean differences of E_{cc}, E_{rr} and E_{ll} between FEM and EFFD were about 3 times higher. These results show that the BEM came closer to the FEM model which we presume to be more physically realistic than the EFFD. EFFD is a powerful tool for modeling deformable objects with arbitrary geometries but it may not be able to fully approximate the physical properties of LV.
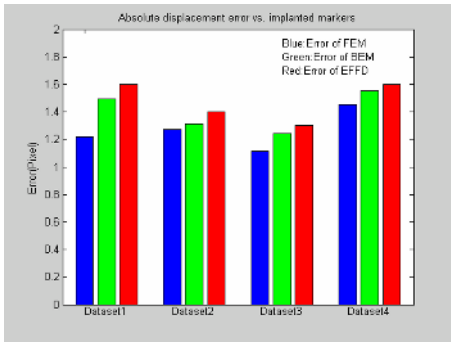

Fig. 3. Absolute displacement error vs. implanted markers. The motion is estimated between ED to ES and compared by the FEM, BEM and EFFD. **Blue**: Error of FEM; **Green**: Error of BEM; **Red**: Error of EFFD
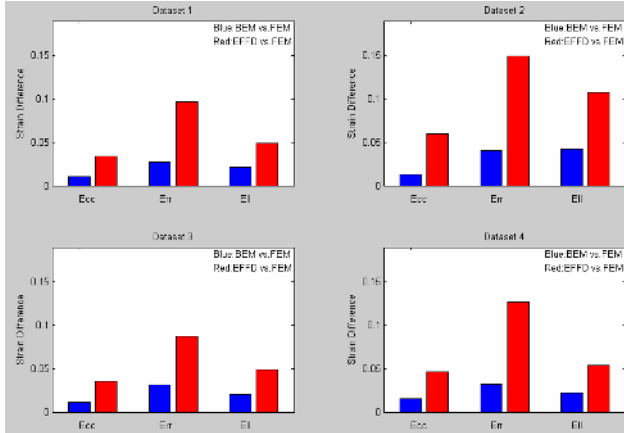

Fig. 4. Comparisons of strains between FEM and BEM, FEM and EFFD: **Blue**: Mean difference between the strain results obtained by FEM and BEM. **Red**: Mean difference between the strain results obtained by FEM and EFFD.

4   Conclusions and Future Work

We conclude that the BEM-based algorithm presented in this paper is both computationally efficient and physically realistic for the following reasons: 1) Only *rough* boundary segmentation is required in the first frame; 2) Only the discretization of surface is required; 3) The fundamental solution matrix only needs to be calculated once in analyzing image sequences; and 4) The deformation is estimated by a biomechanical model of the myocardium. In this paper, by comparing displacement

results using implanted markers, we identified that the FEM has the least errors, while the BEM is the second and EFFD has the larget error. We also found that the difference of strain results using EFFD has larger differences from a FEM approach than the newer BEM-based approach. Further extensions to this work could be using a more complicated biomechanical model, which incorporates the fiber directions of LV.

References

1. P. Shi, A. J. Sinusas, R. T. Constable, E. Ritman, and J. S. Duncan, "Point-tracked quantitative analysis of left ventricular motion from 3D image sequences". *IEEE Transactions on Medical Imaging*, 19(1): 26–50, 2000.
2. J. M. Guccione and A. D. McCulloch. "Finite element modeling of ventricular mechanics", P. J. Hunter, A. McCulloch, and P. Nielsen, editors, *Theory of Heart*, pages 122–144. Springer-Verlag, Berlin, 1991.
3. E. Haber, D. N. Metaxas, and L. Axel. "Motion analysis of the right ventricle from MRI images". *In Medical Image Computing and Computer-Assisted Intervention - MICCAI*, Cambridge, MA, October 1998.
4. F. G. Meyer, R. T. Constable, A. J. Sinusas, and J. S. Duncan. "Tracking myocardial deformation using phase contrast MR velocity fields: A stochastic approach.", *IEEE Transactions on Medical Imaging*, Vol. 15, No. 4, August 1996.
5. K. Kaluzynski, X.Chen, S. Emelianov, A. Skorovoda and M.O'Donnell,"Strain Rate Imaging Using Two-Dimensional Speckle tracking," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, volume 48, 2001: 1111-1123.
6. X.Papademetris, A.J.Sinusas, D. P.Dione, R.T.Constable and J.S.Duncan, "Estimation of 3D Left Ventricular Deformation from Medical Images Using Biomechanical Models," *IEEE Transactions on Medical Imaging*, volume 21, no. 7, July, 2002: 786-799
7. N. Lin and J.S.Duncan, "Generalized Robust Point Matching Using an Extended Free-Form Deformation Model: Application to Cardiac Images," *IEEE International Symposium on Biomedical Imaging (ISBI)*, Arlington, VA, April, 2004
8. S. Coquillart, "Extended free-form deformation: A sculpturing tool for 3D geometric modeling," *Computer Graphics*, vol. 24, no. 4, pp. 187–196, 1990.
9. C.A.Brebbia and J.Dominguez, "Boundary Elements An Introductory Course", 2nd ed, *Computational Mechanics Publications*, McGraw-Hill Book Company
10. N. Lin, X. Papademetris, A. Sinusas, and J.S. Duncan, "Analysis of left ventricular motion using a general robust point matching algorithm," in *Medical Image Computing and Computer-Assisted Intervention*, _Montreal, Canada, November 15-18, 2003, pp. 556–563, (LNCS 2878) Springer Verlag.
11. B.M. Ter, Haar Romeny, L.M.J.Florack,A.H.Salden, and M.A.Viergever, "Higher order differential structure of images", *Image and Vision Computing*, 12(16):317-325,1994
12. E.Trucco and A.Verri., "Introductory Techniques for 3D Computer Vision.", *Prentice Hall*, 1998
13. Wilson, R.B. and Cruse, T.A. (1978), "Efficient implementation of anisotropic three dimensional boundary-integral equation stress analysis", *International Journal for Numerical Methods in Engineering*, 12, 1383-1397