ELASTIC REGISTRATION OF EDGE SETS BY MEANS OF DIFFUSE SURFACES

With an Application to Embedding Purkinje Fiber Networks

Stefan Fürtinger, Stephen Keeling

Institute for Mathematics and Scientific Computing, Karl–Franzens University, Graz, Austria

Gernot Plank, Anton J. Prassl

Institute for Biophysics, Medical University Graz, Graz, Austria

Keywords: Elastic image registration, Edge sets, Diffuse surfaces, Purkinje Fibers, Endocardium.

Abstract: In this work, edge sets are mapped one to the other by representing these zero area sets as *diffuse images* which have positive measure supports that can be registered elastically. The driving application for this work is to map a Purkinje fiber network in the epicardium of one heart to the epicardium of another heart. The approach is to register sufficiently accurate diffuse surface representations of two epicardia and then to apply the resulting transformation to the points of the Purkinje fiber network. To create a diffuse image from a given edge set, a region growing method is used to approximate diffusion of brightness from an edge set to a given point. To be minimized is the sum of squared differences of the registered diffuse images along with a linear elastic penalty for the registration. A Newton iteration is employed to solve the optimality system, and the degree of diffusion is larger in initial iterations while smaller in later iterations so that a desired local minimum is selected by means of vanishing diffusion. Favorable results are shown for registering highly detailed rabbit heart models.

1 INTRODUCTION

The heart is an electrically controlled mechanical pump. It's main function is to drive blood through the circulatory system, thus providing oxygen and metabolites to the organs. A well coordinated electrical activation sequence is of vital importance for allowing an energy-efficient mechanical contraction. In the ventricles, the main pumping chambers of the heart, the electrical impulse is conducted first via the specialized conduction system, referred to as the Purkinje system (PS).

The PS is a highly ramified network of thin cable-like 1D structures which synchronizes ventricular activation by quickly distributing the electrical impulse to the endocardium, i.e. the surfaces of the ventricular cavities. The PS is electrically isolated from the ventricles except at discrete endpoints (Purkinje-myocardial junctions) (Tranum-Jensen et al., 1991). Transmission of the electrical signals at these discrete junctional sites, referred to as Purkinje ventricular junctions (PVJ) is essential to excite the ventricular mass (Huelsing et al., 1998). A loss of electrical syn-

chronicity entails an impairment of the heart's ability to pump blood, which, ultimately, may even lead to sudden cardiac death.

Despite recent advances, PS activity at the organ level cannot be observed directly with currently available experimental modalities. Computer models quite naturally suggest themselves as a surrogate technique to bridge the gap between experimental observations, typically recorded at the epicardial surface of the heart, and electrical events occurring within the PS, at the ventricular epicardium or within the depth of ventricular walls. Despite major recent advancements in modeling technology (Prassl et al., 2009; Plank et al., 2009), integrating topologically realistic models of the PS with anatomically and functionally realistic models of the ventricles remains to be challenging. Owing to the physiological importance of the PS it is highly desirable to include the PS in computational models.

The purpose of the present work was to develop a mathematical framework which enables the mapping of the endocardial PS between different ventricular surface geometries. In the absence of ex-

perimental data, a literature-based PS (Vigmond and Clements, 2007), constructed for the San Diego rabbit heart model (Vetter and McCulloch, 1998), served as reference topology. A recent anatomically highly realistic model of rabbit ventricles (Bishop et al., 2010) served as a target for developing and testing the mapping technique. Both models are shown in Figure 4.1.

We first seek a geometric transformation to match the template heart model to a given reference model. In the context of mathematical image processing this can be seen as 3D registration problem. Once the transformation is found it can be applied to map structures (like the PS) within the template model onto the reference model. This approach guarantees that not only topological features of the PS but also its relative position to the ventricle are preserved and projected onto the reference heart.

Given the sheer size of the models considered here it would require a massive computational effort to calculate only simple transformations. Hence we developed a method that is capable of computing even highly non-linear transformations in a reasonable time while requiring only moderate computational resources. We performed a dimensional reduction of the problem by treating the 3D models as sequence of 2D edges. This strategy reduces memory consumption considerably while simultaneously allowing us to use very efficient techniques to solve the occurring 2D registration problems.

2 EDGES AS BINARY IMAGES

We consider slices arising from cuts through (a) a realistic model of rabbit ventricles (Bishop et al., 2010) and (b) the San Diego rabbit heart (Vetter and McCulloch, 1998). Thus we have two-dimensional edge-sets Γ_0 and Γ_1 and we assume that both edge-sets have finite Hausdorff-measure $\mathcal{H}^1(\Gamma_i) < \infty$ for $i = 0, 1$. Let $\Omega := [1, N]^2 \subset \mathbb{R}^2$ with $N \in \mathbb{N}$ and let I_0 and I_1 be the characteristic functions of Γ_0 and Γ_1 respectively. Then I_0 and I_1 can be interpreted as binary images on Ω. The goal now is to find a displacement $\mathbf{w} : \mathbb{R}^2 \to \mathbb{R}^2$ such that $I_0(\mathbf{x} + \mathbf{w}(\mathbf{x})) \approx I_1(\mathbf{x})$ for all $\mathbf{x} \in \Omega$. For the sake of brevity the argument of $\mathbf{w}$ is omitted in the following.

One approach to the computation of the desired displacement is to treat points on Γ_0 as if connected to one another by elastic springs which are perturbed minimally to meet the target set Γ_1. However, because of the potentially very high computational complexity of such a formulation, the approach used here to match the edge sets is to embed them into images which are then registered elastically. Elastic poten-

tial energy has been used by many authors to regularize image registration; see, e.g., (Peckar et al., 1999), (Keeling and Ring, 2005) and particularly the review in (Modersitzki, 2004). Such regularization is particularly natural when used to register images of tissues having undergone relatively small displacements. However, in the present context, the required displacement field is highly nonlinear, owing partly to the complex geometry of the heart and partly to the great difference in regularity of the two given edge sets. Nevertheless elastic registration is employed here, but with considerable precautions.

In addition to regularization, a notion of image similarity must be selected for the image registration problem at hand. Assuming that $\{I_i\}_{i=0,1} \subset L^2(\Omega)$ one of the simplest distance measures (see for instance (Fitzpatrick et al., 2000)) is the sum of squared intensity differences (SSID) which in this case is given by

$$\frac{1}{2} \int_\Omega |I_0(\mathbf{x} + \mathbf{w}) - I_1(\mathbf{x})|^2 \, d\mathbf{x}. \qquad (2.1)$$

However, in this form the approach is not feasible for the problem: both Γ_0 and Γ_1 are null sets but $\text{supp}(I_i) = \Gamma_i$ for $i = 0, 1$. Hence the trivial deformation $\mathbf{w} \equiv 0$ minimizes (2.1). Another in the context of edges perhaps more natural approach to measure difference is employing the Hausdorff-distance which is a popular tool in computer graphics and image processing and mainly used for shape recognition problems: In (Knauer et al., 2009) a method for minimizing the Hausdorff-distance under translations and rigid motions is developed. Though the proposed algorithm is efficient it is limited to rigid transformations. In (Fuchs et al., 2009) an elastic deformation distance in a shape space is introduced. However, calculating the shortest path between shapes proved to be computationally expensive. Another work (Droske and Ring, 2006) developed a regularized shape gradient descent algorithm within a level-set framework for simultaneous registration and segmentation. However, the present problem still lacks sufficient structure to be treated directly by such approaches.

We present here a technique that combines the simplicity of the SSID-measure (2.1) with the accuracy of the Hausdorff-distance. Instead of looking at I_0 and I_1 directly we consider approximations of edge-sets by diffuse regions in images (a similar approach is presented in (Yang et al., 2008)). Note that this strategy is also employed by Ambrosio and Tortorelli (Ambrosio and Tortorelli, 1990) in their approximation of the Mumford-Shah functional (Mumford and Shah, 1989). Here, for $\varepsilon > 0$ define I_i^ε for $i = 0, 1$ by

$$I_i^{\varepsilon}(\mathbf{x}) := \begin{cases} 1 - d_{\Gamma_i}(\mathbf{x})/\varepsilon, & \text{if } d_{\Gamma_i}(\mathbf{x}) \leq \varepsilon, \\ 0, & \text{otherwise.} \end{cases} \quad (2.2)$$

The nonzero regions of I_i^{ε} are diffuse extensions of the edges Γ_i. Since the distance function $d_{\Gamma_i}(\mathbf{x})$ can be expensive to compute, I_i^{ε} is calculated in practice by a marching procedure in which the distance $d_{\Gamma_i}(\mathbf{x})$ is approximated in terms of the number of marching steps from the edge set Γ_i to $\mathbf{x}$. The extent of this "region-growing" depends on the magnitude of ε: the effect looks like a silk painting of Γ_i where ε affects the duration of the brush touching the fabric. From an image processing point of view this technique can be seen as distance transform. The use of distance transforms in image registration is not new: for instance in (Hill and Baldock, 2006) the authors used constrained distances in the context of interactive non-rigid registration. A variational approach to match distance functions was presented in (Paragios and Ramesh, 2002). Now we may use (2.1) on the region-grown versions I_i^{ε} of I_i to obtain an adapted distance measure

$$S^{\varepsilon}(\mathbf{w}) := \frac{1}{2} \int_{\Omega} |I_0^{\varepsilon}(\mathbf{x} + \mathbf{w}) - I_1^{\varepsilon}(\mathbf{x})|^2 \, d\mathbf{x}. \quad (2.3)$$

As indicated above, we assume that $\mathbf{w}$ is an elastic deformation; thus we define the following linear elastic potential (see for instance (Modersitzki, 2004))

$$E(\mathbf{w}) := \frac{\lambda}{2} \int_{\Omega} (\nabla \cdot \mathbf{w})^2 \, d\mathbf{x} + \frac{\mu}{4} \int_{\Omega} |\nabla \mathbf{w}^{\top} + \nabla \mathbf{w}|_F^2 \, d\mathbf{x}, \quad (2.4)$$

where $|\cdot|_F$ denotes the Frobenius-norm and μ and λ are positive constants describing the elastic properties of the body, the so-called Navier–Lamé constants. Additively extending S^{ε} by the linear elastic potential E, ensures that among all solutions elastic deformations are favored. Hence we end up with the following cost functional

$$J(\mathbf{w}) := S^{\varepsilon}(\mathbf{w}) + E(\mathbf{w}) \quad (2.5)$$

for a fixed $\varepsilon > 0$. Assuming that $\mathbf{w} \in H^1(\Omega)$ the cost J is well defined. Summarized we compute a registration by solving the following minimization problem

$$\min_{\mathbf{w} \in H^1(\Omega)} J(\mathbf{w}). \quad (2.6)$$

Note that the images I_i^{ε} vary with the value of ε. We have proved the existence of a global minimizer for (2.6) for each fixed ε as well as convergence of these minimizers as $\varepsilon \to 0$; however, these proofs are not given here. Furthermore, we note that a global minimizer for ε sufficiently small is essentially $\mathbf{w} = 0$, since (2.1) becomes negligible; thus, we are more interested in the local minimizers introduced by our novel vanishing diffusion strategy (see Algorithm 1 below) which starts with ε large enough that the desired local minimum is dominant.

2.1 Optimality Conditions

We solve problem (2.6) by (formally) deducing the corresponding Euler–Lagrange equations which we linearize using Newton's method (based on the work in (Keeling, 2007)). To obtain the necessary optimality conditions for the optimization problem (2.6) we compute the variational derivative of J in an arbitrary direction $\mathbf{v} \in C^{\infty}(\bar{\Omega})$

$$\frac{\delta J}{\delta \mathbf{w}}(\mathbf{w}; \mathbf{v}) := \frac{d}{ds} J(\mathbf{w} + s\mathbf{v})\big|_{s=0}. \quad (2.7)$$

To keep things clear we employ the linearity of the Gâteaux derivative and split the computation into two parts. Starting with S^{ε} we get

$$\frac{\delta S^{\varepsilon}}{\delta \mathbf{w}}(\mathbf{w}; \mathbf{v}) = \int_{\Omega} (I_0^{\varepsilon}(\mathbf{x} + \mathbf{w}) - I_1^{\varepsilon}(\mathbf{x})) \nabla I_0^{\varepsilon}(\mathbf{x} + \mathbf{w}) \cdot \mathbf{v} \, d\mathbf{x}, \quad (2.8)$$

For $E(\mathbf{w})$ we obtain

$$\frac{\delta E}{\delta \mathbf{w}}(\mathbf{w}; \mathbf{v}) = \int_{\Omega} \mu (\nabla \mathbf{w}^{\top} + \nabla \mathbf{w}) : (\nabla \mathbf{v}^{\top} + \nabla \mathbf{v}) \, d\mathbf{x} \\ + \int_{\Omega} \lambda (\nabla \cdot \mathbf{w})(\nabla \cdot \mathbf{v}) \, d\mathbf{x}, \quad (2.9)$$

where : denotes a component-wise matrix scalar product. The weak necessary condition for the minimization of J is

$$\frac{\delta J}{\delta \mathbf{w}}(\mathbf{w}; \mathbf{v}) = \frac{\delta S^{\varepsilon}}{\delta \mathbf{w}}(\mathbf{w}; \mathbf{v}) + \frac{\delta E}{\delta \mathbf{w}}(\mathbf{w}; \mathbf{v}) = 0, \quad \forall \mathbf{v} \in C^{\infty}(\bar{\Omega}). \quad (2.10)$$

To obtain a strong optimality formulation, we assume first that $\mathbf{w}$ is sufficiently regular. By using partial integration we shift the derivatives from $\mathbf{v}$ to $\mathbf{w}$ and apply the fundamental Lemma of calculus of variations to obtain the Euler–Lagrange equations of (2.6)

$$\begin{cases} \mathcal{E}\mathbf{w} - \mathbf{f}(\mathbf{x}, \mathbf{w}; I_0^{\varepsilon}, I_1^{\varepsilon}) = 0, & \text{in } \Omega, \\ \lambda n_{\ell} \nabla \cdot \mathbf{w} + \mu (\nabla w_{\ell} + \partial_{x_{\ell}} \mathbf{w}) \cdot \mathbf{n} = 0, & \text{on } \partial\Omega, \end{cases} \quad (2.11)$$

where $\mathcal{E}$ is the elasticity operator defined by

$$\mathcal{E}\mathbf{w} := \mu \Delta \mathbf{w} + (\mu + \lambda) \nabla (\nabla \cdot \mathbf{w}), \quad (2.12)$$

and

$$\mathbf{f}(\mathbf{x}, \mathbf{w}; I_0^{\varepsilon}, I_1^{\varepsilon}) := (I_0^{\varepsilon}(\mathbf{x} + \mathbf{w}) - I_1^{\varepsilon}(\mathbf{x})) \nabla I_0^{\varepsilon}(\mathbf{x} + \mathbf{w}), \quad (2.13)$$

is the driving force of the registration. Note that $\mathbf{f}$ is the Gâteaux derivative of the employed distance measure. This gives rise to some important observations: if the force field $\mathbf{f}$ is small a slight deformation $\mathbf{w}$ is enough to satisfy (2.11). Thus the distance measure has to be sensitive enough to "capture" differences in the pathway of the edges Γ_i. Hence a naive application of the SSID-measure on the original binary images I_i is indeed not feasible for our problem: the driving force $\mathbf{f}(\mathbf{x}, \mathbf{w}; I_0, I_1)$ is too small to yield a sufficiently large deformation $\mathbf{w}$ to change the shape of Γ_0 visibly.

However, at the same time the applied distance measure has to be robust enough to "ignore" minor differences in the details of the edges Γ_i in the presence of noise. Though the Hausdorff–distance generates a sufficiently large driving force for an elastic registration it is very sensitive to noise: only modifications of the Hausdorff–distance-measure are capable of matching noisy image data (see e.g. (Sim et al., 1999) or (Zhao et al., 2005)).

The technique of "region-growing" proposed here is able to satisfy both requirements: depending on the choice of $\varepsilon > 0$ the broadening of the blurred edges $\tilde{\Gamma}_i^\varepsilon := \text{supp}(I_i^\varepsilon)$ for $i = 0, 1$ can be arbitrarily extended thus provoking a large driving force $\mathbf{f}(\mathbf{x}, \mathbf{w}; I_0^\varepsilon, I_1^\varepsilon)$ if the original edges Γ_i differ significantly. On the other hand the blurring effect or our "region-growing" approach automatically smooths noisy edges by simultaneously preserving characteristic features. However, an inadequate choice of ε can lead to either excessive blurring and hence loss of important features or insufficient enhancement of the edges and thus a too small driving force $\mathbf{f}$. Though the value of ε is therefore critical the process of finding a suitable ε is usually not too difficult. Nevertheless, in the following we present a solution strategy which is robust against the choice of ε.

2.2 Solution Strategy

Since the Euler–Lagrange equations (2.11) are a system of nonlinear partial differential equations (PDEs) in $\mathbf{w}$ we employ Newton's method on the functional J which takes the form

$$\begin{cases} \dfrac{\delta^2 J}{\delta \mathbf{w}^2}(\mathbf{w}_k; \mathbf{v}, \delta \mathbf{w}_k) = -\dfrac{\delta J}{\delta \mathbf{w}}(\mathbf{w}_k; \mathbf{v}), & \forall \mathbf{v} \in C^\infty(\bar{\Omega}), \\ \mathbf{w}_{k+1} = \mathbf{w}_k + \tau \delta \mathbf{w}_k, \end{cases}$$
(2.14)

where $\tau > 0$ denotes a given step-size and $k = 1, 2, \ldots$ is the iteration index. We introduce the symmetrization

$$\hat{\nabla} I_0^\varepsilon(\mathbf{x}) := \nabla I_0^\varepsilon(\mathbf{x}) \nabla I_0^\varepsilon(\mathbf{x})^\top, \qquad (2.15)$$

and by employing again the linearity of the of the Gâteaux derivative we compute

$$\dfrac{\delta^2 S^\varepsilon}{\delta \mathbf{w}^2}(\mathbf{w}_k; \mathbf{v}, \delta \mathbf{w}_k) = \int_\Omega \mathbf{v} \hat{\nabla} I_0^\varepsilon(\mathbf{x} + \mathbf{w}_k) \delta \mathbf{w}_k \, d\mathbf{x}. \quad (2.16)$$

By using the first variational derivative (2.9) of the linear elastic potential we obtain further

$$\dfrac{\delta^2 E}{\delta \mathbf{w}^2}(\mathbf{w}_k; \mathbf{v}, \delta \mathbf{w}_k) = \int_\Omega \mu(\nabla \mathbf{v}^\top + \nabla \mathbf{v}) : (\nabla \delta \mathbf{w}_k^\top + \nabla \delta \mathbf{w}_k) \\ + \int_\Omega \lambda(\nabla \cdot \mathbf{v})(\nabla \cdot \delta \mathbf{w}_k) \, d\mathbf{x}, \quad (2.17)$$

Using again partial integration and the fundamental Lemma of calculus of variations we arrive (under stronger assumptions on the regularity of $\mathbf{w}$) at the following strong formulation:

$$(-\mathcal{E} + \hat{\nabla} I_0^\varepsilon(\mathbf{x} + \mathbf{w}_k)) \delta \mathbf{w}_k = \mathcal{E} \mathbf{w}_k - \mathbf{f}(\mathbf{x}, \mathbf{w}_k; I_0^\varepsilon, I_1^\varepsilon)),$$
(2.18)

in Ω with boundary conditions

$$\lambda n_\ell \nabla \cdot \mathbf{w} + \mu(\nabla w_\ell + \partial_{x_\ell} \mathbf{w}) \cdot \mathbf{n} = 0, \qquad (2.19)$$

on $\partial\Omega$. Expressions (2.18)-(2.19) form the wanted linear PDE-system for the unknown function $\delta \mathbf{w}_k$. We could prove that the weak form (2.14) of the Newton step admits an unique solution for each k if the image $I_0^\varepsilon(\mathbf{x} + \mathbf{w}_k)$ manifests sufficiently few symmetries. Convergence of Newton's method (2.14) for a fixed $\varepsilon > 0$ and a suitable initial guess can be shown using the Newton–Kantorovich Theorem (see e.g. (Ortega, 1968)).

3 NUMERICAL APPROXIMATION

We will first introduce a discretization scheme for the strong formulation (2.18)-(2.19) of the Newton step and then explain the discrete realization of Newton's method (2.14). We start by rewriting the Euler-Lagrange equations (2.11). Let from now on $\mathbf{w}(\mathbf{x}) := (u(\mathbf{x}), v(\mathbf{x}))^\top$, and $\mathbf{x} = (x, y) \in \Omega$. Then we may rewrite the elasticity operator $\mathcal{E}$

$$\begin{aligned} \mathcal{E} \mathbf{w} &= \mu \left(\dfrac{\partial^2 \mathbf{w}}{\partial x^2} + \dfrac{\partial^2 \mathbf{w}}{\partial y^2} \right) + (\mu + \lambda) \nabla \left(\dfrac{\partial u}{\partial x} + \dfrac{\partial v}{\partial y} \right) \\ &= \begin{pmatrix} (\lambda + 2\mu)\frac{\partial^2}{\partial x^2} + \mu \frac{\partial^2}{\partial y^2} & (\lambda + \mu)\frac{\partial^2}{\partial x \partial y} \\ (\lambda + \mu)\frac{\partial^2}{\partial x \partial y} & \mu \frac{\partial^2}{\partial x^2} + (\lambda + 2\mu)\frac{\partial^2}{\partial y^2} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \\ &=: \begin{pmatrix} \mathcal{E}_{11} & \mathcal{E}_{12} \\ \mathcal{E}_{21} & \mathcal{E}_{22} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}. \end{aligned}$$
(3.1)

For the sake of simplicity we drop the iteration index k here. Since we are working with digital images we define a grid $\Omega_h := \{1, \ldots, N\}^2$, where N denotes the resolution of the images. We use an unit step size $h := 1$, i.e. the width of a cell is one, and employ standard central finite differences to discretize the Newton step (2.18)-(2.19). In particular let $\mathbf{j} := (j_1, \ldots, j_N) \in \mathbb{R}^N$ be an integer component multi index, $\mathbf{1} := (1, \ldots, 1)^\top \in \mathbb{R}^N$ and the cell centroids be given by $\mathbf{x}_j := \mathbf{j}, 1 \leq \mathbf{j} \leq N \cdot \mathbf{1}$. We denote the array arising from evaluating e.g. u at each grid point by $u(\Omega_h) \in \mathbb{R}^{N \times N}$. Hence $U_{\mathbf{j}} \approx u(\mathbf{x}_j)$ and $\vec{u} \in \mathbb{R}^{N^2}$ denotes the vector of values $\{U_{\mathbf{j}}\}$ corresponding to the lexicographic ordering in which j_1 increments first from 1 to N, then j_2 and so on. Further, let $\mathbf{D}(\vec{u}) \in \mathbb{R}^{N^2 \times N^2}$ be the diagonal matrix arising from situating the values $\{U_{\mathbf{j}}\}$ along the diagonal according to lexicographic ordering.

We start by discretizing the elasticity operator $\mathcal{E}$. We show for instance the discretization of $\mathcal{E}_{11}$ near the lower left corner of Ω_h

$$
\begin{pmatrix} \vdots & & \\ 0 & 0 & -2\mu \\ 0 & 8\mu+4\lambda & -4\mu-4\lambda \\ 0 & 0 & -2\mu \\ 0 & 0 & -2\mu \\ 0 & 2\lambda+4\mu & -2\mu-2\lambda \\ 0 & 0 & 0 \end{pmatrix}
\begin{pmatrix} \vdots & & \\ -2\mu & 0 & -2\mu \\ -4\mu-4\lambda & 16\mu+8\lambda & -4\mu-4\lambda \\ -2\mu & 0 & -2\mu \\ -2\mu & 0 & -2\mu \\ -2\mu-2\lambda & 8\mu+4\lambda & -2\mu-2\lambda \\ 0 & 0 & 0 \end{pmatrix}
\quad \cdots
$$
$$\tag{3.2}$$

The upper right block represents the stencil weights for neighbors of a field cell. Similarly the other blocks show for boundary cells the stencil weights for their neighbors. With the same format we represent the stencils of $\mathcal{E}_{12}$:

$$
\begin{pmatrix} \vdots & & \\ 0 & \mu-\lambda & -\mu-\lambda \\ 0 & 0 & 0 \\ 0 & -\mu-\lambda & \mu+\lambda \\ 0 & \mu-\lambda & -\mu-\lambda \\ 0 & \mu+\lambda & -\mu+\lambda \\ 0 & 0 & 0 \end{pmatrix}
\begin{pmatrix} \vdots & & \\ \mu+\lambda & 0 & -\mu-\lambda \\ 0 & 0 & 0 \\ -\mu-\lambda & 0 & \mu+\lambda \\ \mu+\lambda & 0 & -\mu-\lambda \\ \mu-\lambda & 0 & -\mu-\lambda \\ 0 & 0 & 0 \end{pmatrix}
\quad \cdots
\tag{3.3}
$$

The stencils for $\mathcal{E}_{22}$ and $\mathcal{E}_{21}$ are constructed by adequate copying and mirroring of (3.2) and (3.3) respectively. This gives rise to matrices $\mathbf{E}_{k,\ell} \in \mathbb{R}^{N^2 \times N^2}$ with $1 \leq k, \ell \leq 2$ which form the discrete version of the operator $\mathcal{E}$ under lexicographic ordering, i.e.

$$
\mathcal{E}(u(\Omega_h), v(\Omega_h)) \approx \begin{pmatrix} E_{11}\vec{u} + E_{12}\vec{v} \\ E_{21}\vec{u} + E_{22}\vec{v} \end{pmatrix}, \tag{3.4}
$$

so that we may write

$$
\mathbf{E}\vec{w} := \begin{pmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{pmatrix} \begin{pmatrix} \vec{u} \\ \vec{v} \end{pmatrix}. \tag{3.5}
$$

Now we develop the discretization of the images I_i^ε. The discrete version $\tilde{I}_1 \in \mathbb{R}^{N \times N}$ of I_1^ε is readily established by setting $\tilde{I}_{1,\mathbf{j}} := I_1^\varepsilon(\mathbf{x}_j)$. Analogously we set $\tilde{I}_{0,\mathbf{j}} \approx I_0^\varepsilon(\mathbf{x}_j + \mathbf{w}_j)$. Note that for non-integer values of $\mathbf{w}$ a (multi)linear interpolation scheme has to be applied to compute $I_0^\varepsilon(\mathbf{x} + \mathbf{w})$ and $\nabla I_0^\varepsilon(\mathbf{x} + \mathbf{w})$. We use bilinear interpolation to compute $\tilde{I}_{0,\mathbf{j}}$ and introduce the matrix $\tilde{I}_0 \in \mathbb{R}^{N \times N}$ of values $\{\tilde{I}_{0,\mathbf{j}}\}$ (if $\mathbf{x} + \mathbf{w}$ lies outside of Ω we assign the extrapolation value zero). Finally we approximate the gradient ∇I_0^ε. Let δh denote a fraction of the cell width h, i.e. $\delta h := h/2 = 1/2$. Employing this we get the increment of $I_0^\varepsilon(\mathbf{x_j} + \mathbf{w}_j)$ in x-direction to compute the following discrete derivative

$$
\frac{\partial}{\partial x} I_0^\varepsilon(\mathbf{x_j} + \mathbf{w}_j) \approx \frac{1}{2\delta h} \left(I_0^\varepsilon(\mathbf{x_j} + e_1 \delta h + \mathbf{w}_j) \right. \tag{3.6}
$$
$$
\left. - I_0^\varepsilon(\mathbf{x_j} - e_1 \delta h + \mathbf{w}_j) \right) =: \delta_x \tilde{I}_{0,\mathbf{j}},
$$

where e_1 is the first unit vector in $\mathbb{R}^2$. Note that we use again bilinear interpolation if I_0^ε is evaluated at

non-grid points. Analogously we compute $\delta_y \tilde{I}_{0,\mathbf{j}}$ and are finally ready to set up the discrete Newton step. With : denoting again a componentwise matrix scalar product let

$$
\mathbf{G}(\tilde{I}_0) := \begin{pmatrix} D\left(\overrightarrow{\delta_x \tilde{I}_0 : \delta_x \tilde{I}_0}\right) & D\left(\overrightarrow{\delta_x \tilde{I}_0 : \delta_y \tilde{I}_0}\right) \\ D\left(\overrightarrow{\delta_x \tilde{I}_0 : \delta_y \tilde{I}_0}\right) & D\left(\overrightarrow{\delta_y \tilde{I}_0 : \delta_y \tilde{I}_0}\right) \end{pmatrix} \in \mathbb{R}^{2N^2 \times 2N^2},
\tag{3.7}
$$

and define

$$
\vec{f} := \begin{pmatrix} \overrightarrow{(\tilde{I}_0 - \tilde{I}_1) : \delta_x \tilde{I}_0} \\ \overrightarrow{(\tilde{I}_0 - \tilde{I}_1) : \delta_y \tilde{I}_0} \end{pmatrix} \in \mathbb{R}^{2N^2}. \tag{3.8}
$$

Then the Newton step (2.18)-(2.19) is discretized by

$$
\left(-\mathbf{E} + \mathbf{G}(\tilde{I}_0)\right) \delta\vec{w} = -\mathbf{E}\vec{w} - \vec{f}, \tag{3.9}
$$

which is a linear equation system in the unknown $\delta\vec{w} \in \mathbb{R}^{2N^2}$

3.1 The Discrete Newton Iteration

Now we want to establish a discrete version of Newton's method (2.14) for the functional J. We use $\vec{w}_1 := 0 \in \mathbb{R}^{2N^2}$ as initial guess and employ the discrete Newton step (3.9) to obtain the following iteration

$$
\begin{cases} \left(-\mathbf{E} + \mathbf{G}(\tilde{I}_{0,k})\right) \delta\vec{w}_k = -\mathbf{E}\vec{w}_k - \vec{f}_k, \\ \vec{w}_{k+1} = \vec{w}_k + \tau_k \delta\vec{w}_k, \end{cases} \quad k = 1, 2, \ldots
\tag{3.10}
$$

Note that the discrete gradient $\mathbf{G}(\tilde{I}_{0,k})$ and the discrete force $\vec{f}_k$ depend on the current value of $\vec{w}_k$ hence they have to be updated at each iteration.

We use a backtracking-like line search to determine the step size τ_k. We establish a discrete approximation J_h of the cost functional J and determine τ_k as follows:

$$
\begin{cases} \tau_k = \min_{\tau \in \mathcal{T}} J_h(\vec{w}_k + \tau\delta\vec{w}_k), \\ \mathcal{T} := \left\{ \tau = \frac{2\ell}{L} \mid \ell = 1, \ldots, L \right\}, \end{cases}
\tag{3.11}
$$

where $\vec{w}_k$ denotes the current iterate and $\delta\vec{w}_k$ the currently computed Newton direction. Note that this line-search algorithm determines the optimal step-size on the interval $[2/L, 2]$ hence in contrary to classic backtracking approaches (Dennis and Schnabel, 1983) step sizes larger than one can be chosen. This method has proven to provide good performance and less total computational cost than standard Armijo–Goldstein or Wolfe–Powell techniques (Nocedal and Wright, 2000). Since $L > 0$ is not chosen to be particularly large only few evaluations of J_h are computed to obtain τ_k and the computation of J_h does not involve the expensive calculation of the discrete gradient $\mathbf{G}(\tilde{I}_{0,k})$.

We use a relative stopping criterion in the Newton iteration (3.10). Note that the right hand side of the Newton step (3.9) corresponds to the discretized Euler–Lagrange equations (2.11) of the original minimization problem (2.6). Hence we want the residual $r_k := -E\vec{w}_k - \vec{f}_k$ to be "small" and thus define the relative residual $r_b := |r_k|/|r_1|$. Additionally we compute the relative change of the iterates $r_e := |\vec{w}_k - \vec{w}_{k-1}|/|\vec{w}_k|$ (where $r_e := 0$ if $|\vec{w}_k| = 0$) and combine these two notions in a stopping criterion.

As mentioned above the choice of ε may have negative effects on the outcome of the registration. Whereas it is usually not too difficult to avoid too small values of ε it is very hard to give an upper bound for ε beforehand. However, a value of ε too large leads to excessively blurred approximations I_i^ε of I_i and hence a loss of potentially important features of the original edges Γ_i. Hence we want the solution strategy of the registration problem (2.6) to be robust against the choice of ε. To account for the introduced approximation error we augment Newton's method (3.10) with an outer iteration in which we reduce blurring. We compute a rough solution $\vec{w}^\star$ of (3.10) by using a low number of maximal iterations $k_{\max}$. To reduce blurring in the images we compute the element-wise-squared of $\tilde{I}_i^\varepsilon$ which accentuates the original edge sets Γ_i in contrast to their blurred surroundings. Then we restart Newton's method with the images $\tilde{I}_i^\varepsilon(\mathbf{x}_j)^2$, $\vec{w}_1 = \vec{w}^\star$ and increase $k_{\max}$. We repeat this procedure K times such that the squared images are sufficiently close approximations to the original images, i.e $\tilde{I}_i^\varepsilon(\mathbf{x}_j)^{2^K} \approx I_i(\mathbf{x}_j)$. To fix ideas we summarize this method in Algorithm 1.

Algorithm 1: Iterative method to solve the elastic registration problem (2.6).

Choose $\varepsilon > 0$, $k_{\mathrm{inc}} \in \mathbb{N} : k_{\mathrm{inc}} \geq 2$, tol > 0 and $\vec{w} \in \mathbb{R}^{2N^2}$.

1. Given the edge-sets Γ_0 and Γ_1 embed them in the center of images I_0 and I_1.

2. Compute diffuse versions I_i^ε of I_i.

3. `for` $\kappa = 1$ `to` K

   (a) Set $k_{\max} = k_{\mathrm{inc}} \cdot \kappa$
   (b) Compute $\vec{w}^\star$ according to (3.10) using the line search strategy (3.11) until either $\min(r_b, r_e) <$ tol or $k >$ $k_{\max}$.
   (c) Set $\tilde{I}_i^\varepsilon(\mathbf{x}_j) := \tilde{I}_i^\varepsilon(\mathbf{x}_j)^2$ and $\vec{w}_1 = \vec{w}^\star$.

4. `end`

5. Set $\vec{w} = \vec{w}^\star$.

The centering of the edges as stated in Algorithm 1 is an important pre-registration step: if the edges are not centered within the images I_i the diffuse extensions of Γ_i created by (2.2) may intersect image boundaries. This would greatly impair the computation of the deformation $\vec{w}$ at the boundary since mass having diffused over the boundary cannot be matched with its natural counterpart in the other image. Moreover, if both edges Γ_0 and Γ_1 share roughly the same area of the images I_0 and I_1 respectively the deformation field does not have to account for large translations within the image domain.

Altogether this approach has proven itself in practice: since the minimization problem (2.6) is only approximately solved for blurred versions $\tilde{I}_i^\varepsilon$ of I_i in the beginning, a rough initial deformation field is obtained that captures only global deformations in I_0. By successively deblurring the images $\tilde{I}_0^\varepsilon$ and $\tilde{I}_1^\varepsilon$ the pathway of the original edge sets Γ_i takes shape again, and the deformation $\vec{w}$ is updated to capture local features of the edges.

4 COMPUTATIONAL RESULTS

All computations were carried out in MATLAB™ 2009b running on a Dell Optiplex 745 equipped with 8 GB of RAM. The used operating system was openSUSE 11.2 (64bit, kernel 2.6.31.12-0.2).

We want to apply the solution strategy developed in the previous section to register the San Diego rabbit heart (Vetter and McCulloch, 1998) (referred to as "TBunnyC" in the following) to an anatomically highly realistic model (Bishop et al., 2010) (called "Oxford-heart" from now on). Both models are shown in Figure 4.1. The basic idea is to slice up the 3D models which gives rise to 2D edges. Then each slice is registered consecutively by employing Algorithm 1. The registered slices are assembled again to obtain a 3D representation of the results. Finally the deformation fields computed for each slice are used to map a literature-based Purkinje fiber network (Vigmond and Clements, 2007) onto the endocardium of the Oxford-heart.

The measuring unit of the models is μm; hence, we rescaled the large values in the models to reduce the size of the arising linear systems in the registration. Further, to obtain a maximal spatial "overlap" we translated the models. Then we "cut" the 3D models to obtain 2D edges. To ensure a clear differentiation between left and right cavity we separate the 3D heart models accordingly. We chose the z-axis as cutting direction which seems to be a natural choice once left and right cavities are considered separately. Due to the fact that the grid points are not equally distributed on regular xy-planes in vertical direction we have to divide the models into layers of finite thickness, particularly with a thickness of 250μm. The

(a)   (b)

Figure 4.1: The given 3D heart models: (a) The San Diego rabbit heart (Vetter and McCulloch, 1998) "TBunnyC" (29 394 points) and (b) the anatomically highly realistic model (Bishop et al., 2010) "Oxford Heart" (258 178 points).
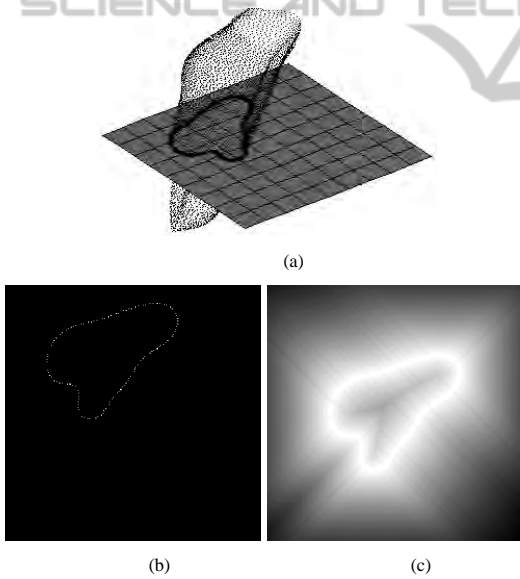

(a)

(b)   (c)

Figure 4.2: Illustration of the dissection of the 3D heart models. Shown is the left cavity of the TBunnyC model. First the model is cut in horizontal direction (a) to generate a 2D edge set (b). We center the image and apply our region-growing algorithm (c).

wanted 2D edges arise from shrinking the thickness until the slice becomes an xy-plane.

Having generated a series of slices we are ready to apply Algorithm 1. Each slice is a sparse double MATLAB matrix of size $N \times N$. We apply two preprocessing steps: for each slice we first center the edges within the images and second we employ our "region-growing" algorithm to obtain blurred ap-

proximations I_i^{ε}. We approximate the distance function d_{Γ_i} used in the definition (2.2) of I_i^{ε} by the following marching scheme: Each non-zero entry of I_i is multiplied by N and added to its 3×3 neighborhood. This procedure is repeated until either the edge set "outgrows" the image or a maximal number of iterations has been exceeded. An exemplary result of this region-growing is depicted in Figure 4.2(c). This way we generated two image series: the blurred slices of the 3D model TBunnyC $\left\{ I_0^{\varepsilon,m} \right\}_{m=1}^M$ which serve as template image stack and the blurred cuts of the Oxford-heart $\left\{ I_1^{\varepsilon,m} \right\}_{m=1}^M$ which form the reference stack. Thus we want to find elastic deformations $\mathbf{w}^m$ so that $I_0^{\varepsilon,m}(\mathbf{x} + \mathbf{w}^m) \approx I_1^{\varepsilon,m}(\mathbf{x})$ for $m = 1, \ldots, M$. This is achieved by employing a MATLAB-implementation of the augmented Newton method depicted in Algorithm 1. A list of all used parameters and their values is given in Table 4.1. The discrete Newton step (3.9) is solved using the inbuilt MATLAB-function `mldivide` ("backslash"). To utilize memory efficiently the coefficient matrices in each Newton step are sparse double MATLAB matrices. Finally we apply the computed deformations $\mathbf{w}^m$ to the original images I_0^m to register the edges Γ_0^m to Γ_1^m. Figure 4.3 sketches the procedure. Figure 4.4 shows the 3D reconstruction of the registered TBunnyC slices in comparison to the 3D reference, i.e. the Oxford heart, for both left and right cavities.

Having registered each slice of the left and right cavities we want to utilize the computed deformation fields to map an artificial Purkinje fiber network given for the TBunnyC model onto the Oxford heart. The

18

I_0^ε (a) $I_0^\varepsilon(\mathbf{x}+\mathbf{w})^{2^K}$ (b) $I_0(\mathbf{x}+\mathbf{w})$ (c)

I_1^ε (d) $I_1^\varepsilon(\mathbf{x})^{2^K}$ (e) $I_1(\mathbf{x})$ (f) $\mathbf{w}(\mathbf{x})$ (g)
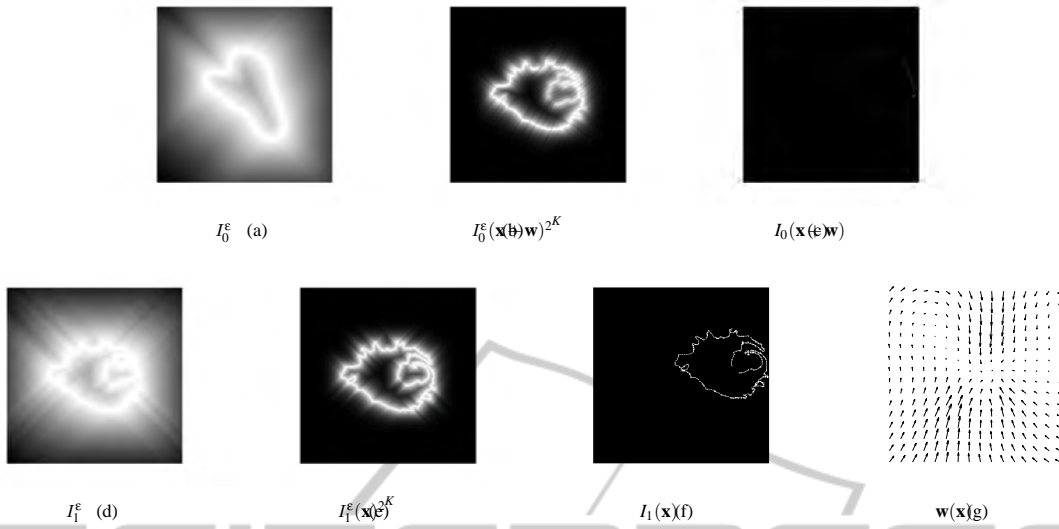
Figure 4.3: The different stages of Algorithm 1 for the template image I_0 (top row) and the reference image I_1 (bottom row). Panels (a) and (d) show the centered and region grown versions of the images. Panel (b) presents the resulting image $I_0^\varepsilon(\mathbf{x}+\mathbf{w})^{2^K}$ after completion of Algorithm 1 versus the final reference $I_1^{\varepsilon 2^K}(\mathbf{x})$ (e). Panel (c) depicts the original image I_0 after application of the computed transformation $\mathbf{w}$ versus the original target I_1 (f). Panel (g) shows the computed deformation field $\mathbf{w}(\mathbf{x})$.

Table 4.1: The used parameters.

Parameter	Value	Meaning
λ	1e-2	see (2.4)
μ	1e-2	see (2.4)
L	10	see (3.11)
tol	1e-3	see Algorithm 1
K	4	see Algorithm 1
k_{inc}	5	see Algorithm 1

PS was given as list of Cartesian coordinates of 832 points (414 points for the left, 418 points for the right cavity respectively) representing the spatial locations of the nodes of a 3D graph. Figure 4.5 shows the Purkinje fiber network in the TBunnyC model in comparison to the registered network in the Oxford heart.

5 DISCUSSION AND CONCLUSIONS

The artificial PS considered here was modeled to be a subset of the TBunnyC model. Hence we may safely assume that the computed deformations constitute a suitable mapping for projecting the network nodes from the TBunnyC model onto the Oxford heart. The application required the registered network to be a subset of the Oxford heart. This requirement together with the sheer number of network nodes rendered it impossible to construct an affine linear mapping to roughly project the nodes into a proximity of

the Oxford endocardium and successively correcting the spatial position of each individual point. Moreover, the complex geometry of the Oxford heart which differs significantly from the topology of the TBunnyC model proved to be captured adequately only by elastically deforming the TBunnyC endocardial walls. Hence the computed elastic deformations guarantee that despite even large differences in the endocardial geometries of both models the artificial Purkinje fiber network is mapped sufficiently close to the Oxford endocardium. Post-processing can be used to project nodes onto the nearest facet.

Though our results have been positively evaluated by experts, the lack of experimental observations makes a rigorous validation difficult. While our 2D approach depends strongly upon the preregistrations performed in 3D and in 2D, this dependence might be relaxed by a full 3D registration at considerably higher cost. Our simulations confirmed that separating the cavities of the heart models is crucial. This can be seen rather easily by looking at Figure 4.1: we see that cutting the hearts in vertical direction gives rise to slices which contain edges from both the left and right cavities. This fact seriously impairs the outcome of the registration: depending on the choice of the Navier–Lamé constants and the proximity of the cavities the computed deformation fields "pull" edges corresponding to the left cavity to an edge arising from a cut through the right cavity and vice versa. Despite numerous tests using different parameter values and
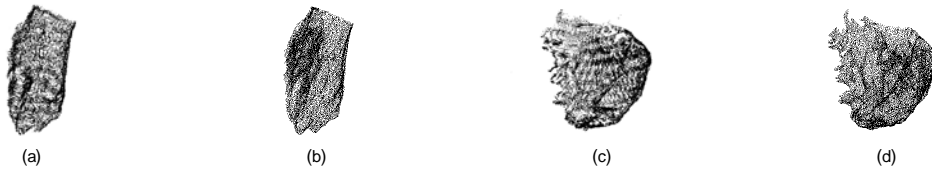
Figure 4.4: 3D Reconstruction of the registered TBunnyC slices for the left (a) and right (c) cavity vs the left (b) and right (d) cavity of the reference Oxford heart.
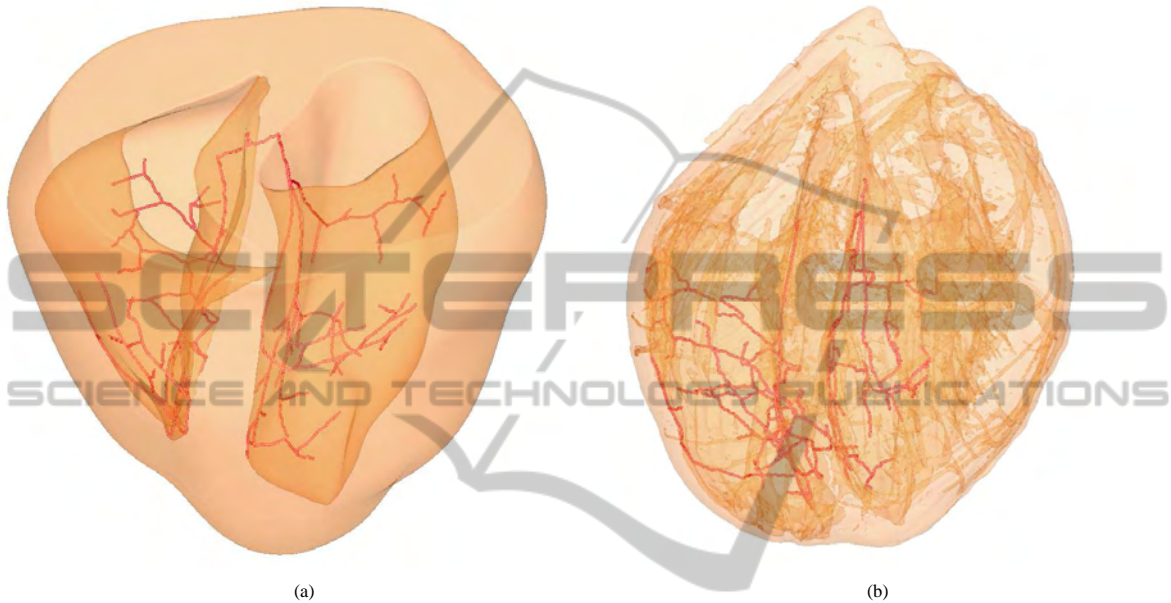

Figure 4.5: The artificial Purkinje fiber network in the TBunnyC model (a) and the registered network in the Oxford heart (b).

cutting directions this undesirable effect could not be completely eradicated. Hence we separated the 3D hearts into left and right cavities.

Despite the application presented here our method proved to be a highly efficient and reliable technique to register 2D edges. In contrast to previously developed techniques involving the Hausdorff-distance our approach is computationally cheap and not limited to rigid transformations. The approximation of edges by diffuse surfaces allows us to use a standard SSID distance-measure which is simple to compute and can be easily extended by an elastic penalizer to account for nonlinear deformations. Due to the plain structure of the associated cost functional the derivation of necessary optimality conditions by means of variational calculus is straight forward. The use of variational derivatives further enables us to employ fast and theoretically well-founded optimization routines such as Newton's method. Furthermore, the driving force of the registration can be quickly evaluated and is easy to interpret. In relation to other works employing blurring strategies in registration problems we note that our novel vanishing diffusion strategy described

in Algorithm 1 ensures robustness of the proposed method.

ACKNOWLEDGEMENTS

REFERENCES

Ambrosio, L. and Tortorelli, V. (1990). Approximation of functionals depending on jumps by elliptic functionals via γ-convergence. *Communications on Pure and Applied Mathematics*, 43:999–1036.

Bishop, M. J., Plank, G., Burton, R. A., Schneider, J. E., Gavaghan, D. J., Grau, V., and Kohl, P. (2010). Development of an anatomically detailed MRI-derived rabbit ventricular model and assessment of its impact

on simulations of electrophysiological function. *Am J Physiol Heart Circ Physiol*, 298(2):H699–718.

Dennis, J. and Schnabel, R. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice–Hall, Englewood Cliffs, NJ,.

Droske, M. and Ring, W. (2006). A mumford-shah level-set approach for geometric image registration. *SIAM Journal of Applied Mathematics*, 66(6):2127–2148.

Fitzpatrick, J., Hill, D., and Maurer, C. (2000). *Image Registration, Medical Image Processing*, volume 2, chapter 8 of the Handbook of Medical Imaging. SPIE Press.

Fuchs, M., Jüttler, B., Scherzer, O., and Yang, H. (2009). Shape metrics based on elastic deformations. *J. Math. Imaging Vis.*, 35(1):86–102.

Hill, W. and Baldock, R. A. (2006). The constrained distance transform: Interactive atlas registration with large deformations through constrained distances. In *DEFORM'06 - Workshop on Image Registration in Deformable Environments*.

Huelsing, D. J., Spitzer, K. W., Cordeiro, J. M., and Pollard, A. E. (1998). Conduction between isolated rabbit purkinje and ventricular myocytes coupled by a variable resistance. *Am J Physiol*, 274(4 Pt 2):H1163–H1173.

Keeling, S. and Ring, W. (2005.). Medical image registration and interpolation by optical flow with maximal rigidity. *Journal of Mathematical Imaging and Vision*, 23(1):47–65.

Keeling, S. L. (2007). Generalized rigid and generalized affine image registration and interpolation by geometric multigrid. *Journal of Mathematical Imaging and Vision*, 29:163–183.

Knauer, C., Kriegel, K., and Stehn, F. (2009). Minimizing the weighted directed hausdorff distance between colored point sets under translations and rigid motions. In *FAW '09: Proceedings of the 3d International Workshop on Frontiers in Algorithmics*, pages 108–119, Berlin, Heidelberg. Springer-Verlag.

Modersitzki, J. (2004). *Numerical Methods for Image Registration*. Oxford Science Publications.

Mumford, D. and Shah, J. (1989). Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685.

Nocedal, J. and Wright, S. (2000). *Numerical Optimization*. Springer.

Ortega, J. M. (1968). The newton–kantorovich theorem. *The American Mathematical Monthly*, 75(6):658–660.

Paragios, N. and Ramesh, M. R. V. (2002). Matching distance functions: A shape-to-area variational approach for global-to-local registration. In *Proceedings of the 7th European Conference on Computer Vision-Part II*.

Peckar, W., Schnörr, C., Rohr, K., and Stiehl, H. S. (1999). Parameter-free elastic deformation approach for 2d and 3d registration using prescribed displacements. *J. Math. Imaging Vis.*, 10(2):143–162.

Plank, G., Burton, R. A., Hales, P., Bishop, M., Mansoori, T., Bernabeu, M. O., Garny, A., Prassl, A. J., Bollensdorff, C., Mason, F., Mahmood, F., Rodriguez, B., Grau, V., Schneider, J. E., Gavaghan, D., and Kohl, P. (2009). Generation of histo-anatomically representative models of the individual heart: tools and application. *Philos Transact A Math Phys Eng Sci*, 367(1896):2257–92.

Prassl, A. J., Kickinger, F., Ahammer, H., Grau, V., Schneider, J. E., Hofer, E., Vigmond, E. J., Trayanova, N. A., and Plank, G. (2009). Automatically generated, anatomically accurate meshes for cardiac electrophysiology problems. *IEEE Trans Biomed Eng*, 56(5):1318–30.

Sim, D.-G., Kwon, O.-K., and Park, R.-H. (1999). Object matching algorithms using robust hausdorff distance measures. *IEEE Transactions on Image Processing*, 8(3):425–429.

Tranum-Jensen, J., Wilde, A. A., Vermeulen, J. T., and Janse, M. J. (1991). Morphology of electrophysiologically identified junctions between purkinje fibers and ventricular muscle in rabbit and pig hearts. *Circ Res*, 69(2):429–437.

Vetter, F. and McCulloch, A. (1998). Three-dimensional analysis of regional cardiac function: a model of rabbit ventricular anatomy. *Prog Biophys Mol Biol*, 69(2-3):157–83.

Vigmond, E. J. and Clements, C. (2007). Construction of a computer model to investigate sawtooth effects in the purkinje system. *IEEE Trans Biomed Eng*, 54(3):389–399.

Yang, S., Kohler, D., Teller, K., Cremer, T., Le Baccon, P., Heard, E., Eils, R., and Rohr, K. (2008). Nonrigid registration of 3-d multichannel microscopy images of cell nuclei. *IEEE Transactions on Image Processing*, 17:493–499.

Zhao, C., Shi, W., and Deng, Y. (2005). A new hausdorff distance for image matching. *Pattern Recognition Letters*, 26(5):581–586.