

Efficient and generalizable statistical models of shape and appearance for analysis of cardiac MRI

Alexander Andreopoulos*, John K. Tsotsos

York University, Department of Computer Science and Engineering, Centre for Vision Research, Toronto, Ontario, Canada M3J 1P3

Received 7 July 2006; received in revised form 1 December 2007; accepted 21 December 2007

Available online 11 January 2008

Abstract

We present a framework for the analysis of short axis cardiac MRI, using statistical models of shape and appearance. The framework integrates temporal and structural constraints and avoids common optimization problems inherent in such high dimensional models. The first contribution is the introduction of an algorithm for fitting 3D active appearance models (AAMs) on short axis cardiac MRI. We observe a 44-fold increase in fitting speed and a segmentation accuracy that is on par with Gauss–Newton optimization, one of the most widely used optimization algorithms for such problems. The second contribution involves an investigation on hierarchical 2D + time active shape models (ASMs), that integrate temporal constraints and simultaneously improve the 3D AAM based segmentation. We obtain encouraging results (endocardial/epicardial error 1.43 ± 0.49 mm/ 1.51 ± 0.48 mm) on 7980 short axis cardiac MR images acquired from 33 subjects. We have placed our dataset online, for the community to use and build upon.
© 2008 Elsevier B.V. All rights reserved.

Keywords: Active appearance models; Active shape models; Cardiac MRI; Segmentation

1. Introduction

1.1. Purpose

In 2004, cardiovascular disease (CVD) contributed to almost one third of global deaths (American Heart Association, 2004). CVD is the leading cause of death in the developed world and by 2010, CVD is estimated to be the main cause of death in developing countries. According to 2001 estimates, if all forms of CVD in the United States were eliminated, the average life expectancy would increase by around 10 years (Frangi et al., 2001). An elimination of all forms of cancer, on the other hand, would cause the average life expectancy to increase by 3 years.

Three-dimensional imaging of the heart using imaging modalities such as ultrasound, magnetic resonance imaging (MRI) and X-ray computed tomography is a rapidly developing area of research in medical imaging. Screenings that detect problems at an early stage, when treatment is most effective, can help prevent heart disease. The manual segmentation of short axis cardiac MRI provides clinically useful indicators of heart function, such as the ejection fraction (EF) ratio (Frangi et al., 2001). However, manual segmentation is a slow and error prone procedure and fully automated methods are highly desirable.

There are various reasons why the algorithms described in the literature for the segmentation and functional analysis of cardiac images exhibit lower success rates than that of human experts (Bosch et al., 2002; Gering, 2003; Mitchell et al., 2001, 2002):

1. Existing methods do not incorporate a sufficient amount of *a priori* knowledge about the segmentation problem.
2. Most existing methods do not consider the three-dimensional context as an integral part of their functionality.

* Corresponding author. Tel.: +1 416 736 2100x33257; fax: +1 416 736 5872.

E-mail addresses: alekos@cse.yorku.ca (A. Andreopoulos), tsotsos@cse.yorku.ca (J.K. Tsotsos).

URLs: <http://www.cse.yorku.ca/~alekos/> (A. Andreopoulos), <http://www.cse.yorku.ca/~tsotsos/> (J.K. Tsotsos).

3. Most existing methods do not consider the temporal context as an integral part of their functionality.
4. The algorithm needs to be applicable for a wide variety of image situations. The variability in the images is due to patient health, patient movement and image noise and artifacts.

The following general observations can be made about model-based approaches for the functional analysis of cardiac images. Four-dimensional (4D) cardiac models – which incorporate knowledge about the 3D structure of the heart and its temporal deformation – that rely heavily on training data to build the model, suffer from the *curse of dimensionality* problem: a 4D cardiac model that is capable of generalizing and reliably segmenting the full spectrum of normal and abnormal cardiac deformations requires a considerable number of training samples. Furthermore, as the dimensionality of the model increases, so does the number of parameters which affect the model's deformation. The speed, accuracy and reliability of optimization algorithms is negatively affected when dealing with models that use hundreds of parameters.

At the same time, models which incorporate prior knowledge about the 3D cardiac structure and its temporal deformation are necessary if we are to achieve reliable segmentation with minimal user input. It is not uncommon to observe cardiac sequences where imaging induced artifacts make it *impossible* to segment certain frames of the sequence without prior knowledge of both the temporal deformation and 3D cardiac structure. Models which incorporate such prior knowledge can make reasonable assumptions regarding the segmentation and cause the algorithm's performance to degrade gracefully in cases where non-model-based segmentation algorithms would fail catastrophically.

Models which rely on object appearance – such as active appearance models (AAMs) – are quite robust when dealing with images that are degraded due to noise and other artifacts. With short axis cardiac MRI, the slices closest to the apex tend to have blurred myocardial borders making edge-based segmentation methodologies inappropriate. Prior knowledge about the 3D cardiac structure and appearance can further improve the reliability of the segmentation.

In contrast, basal slices tend to have sharper myocardial borders, making edge-based approaches for model fitting more appropriate. Furthermore, various image structures are often better delineated using edge-based methods. However, edge-based approaches tend to be quite unreliable when not initialized close to the desired boundary.

As the above brief discussion suggests, there is currently no approach to cardiac segmentation that can deal with the full spectrum of image variability. This makes it worthwhile to investigate the integration of various approaches for dealing with the cardiac segmentation problem.

In this paper, we tackle these issues by combining and extending various approaches based on statistical models of shape and appearance (Cootes and Taylor, 2004). We

incorporate knowledge about typical cardiac structure by using a 3D active appearance model (AAM) and a 2D + time active shape model (ASM) of the heart to segment the left ventricle (Frangi et al., 2001). The first model we use is a 3D AAM which encodes the typical deformation of the left ventricle along the x -, y - and z -axis. This model incorporates knowledge about cardiac structure and, thus, provides a robust methodology for performing the segmentation (Mitchell et al., 2001, 2002). We use this model to obtain an initial segmentation from each frame of a data set of cardiac MRI.

Once we have this initial segmentation, we use the 2D + time ASM to improve the segmentation. This second model encodes knowledge about the typical temporal deformation of the heart. More specifically, the second model encodes knowledge about the typical deformation along the x , y , and temporal axis of the cardiac cycle. The relative accuracy and robustness provided by the 3D AAM is used to provide an accurate initialization for the 2D + time ASM. This allows us to use edge-based approaches to further refine the segmentation. We apply this model at various heights of our MRI sequence (along the long axis/ z -axis of the heart) in order to improve the segmentation accuracy.

The contributions presented in the paper are the following. Firstly, we introduce a method for optimizing the 3D AAM used in the first stage of the model fitting. Brute force non-linear optimization using techniques such as Gauss–Newton optimization is accurate and reliable but extremely slow. Faster methods described in the literature are theoretically unsound and often unreliable. To this end, we developed an efficient, robust and theoretically sound algorithm for optimizing 3D active appearance models, the accuracy of which is on par with Gauss–Newton optimization and which runs around 44 times faster. We use various extensions of the inverse compositional algorithm (Baker and Matthews, 2001; Matthews and Baker, 2004) to accomplish this.

The second contribution involves an approach for handling the problem of training 3D ASMs in the presence of small training sets. We accomplish this through the use of wavelets. More specifically, ASMs are often limited by the inability of a small training set to capture the full range of biological shape variability. Using an approach that was inspired by Davatzikos et al. (2003), we use the statistical properties of the wavelet transform of our shape, to make the shape deformation more generalizable and, thus, get better results in the presence of small training sets. This enables us to have robust ASMs which are not as dependent on the training set size and which can improve the segmentation provided by the 3D AAM.

The dataset of short axis cardiac MR images that we use was provided by the Department of Diagnostic Imaging of the Hospital for Sick Children in Toronto, Canada. The dataset and the ground truth of manual segmentations is provided online, for the medical imaging community to use and build upon.

In the following subsection, we present an overview of previous work on cardiac models that are applicable to the segmentation of cardiac MRI. In Section 2, we present our inverse compositional-based approach for rapid and robust fitting of 3D AAMs and we also present the wavelet based hierarchical 2D + time ASM that we use to further improve our results. In Section 3, we present a detailed description of our experimental setup. In Section 4, we provide results demonstrating the validity of our approach. In Section 5, we present a critical assessment of the method and provide suggestions for future research. In Section 6, we conclude the paper.

1.2. Background

We are interested in cardiac models because they incorporate expert knowledge about cardiac structure and its temporal deformation, thus offering a robust method for estimating various indices of cardiac performance. In this section, we provide a brief overview of some of the more distinctive work that has been done on cardiac image analysis using 3D models. This will provide an overview of the main approaches for cardiac modeling.

We limit our exposition to cardiac models built by the medical imaging and computational vision communities which satisfy two criteria. The first criterion is that the models and any shape reconstructed by such models is in 3D or 4D (incorporating temporal properties of the heart). The second criterion is that such models can be applied to cardiac MR images as a means of extracting parameters to interpret cardiac performance. This can include the segmentation of parts of the heart or the motion analysis of the heart. For an introduction to cardiac models built by the cardiology community for simulating cardiac electrophysiology, the reader is referred to Panfilov and Holden (1996). With the exception of the work by Sermesant et al. (2003), and the Yale group (Papademetris et al., 2002; Papademetris, 2000; Yan et al., 2005) which we briefly describe below, most models that have been built by the medical imaging/computer vision community incorporate only limited aspects of the physiology of the heart. We believe that greater interaction between these two communities can bring significant advances to the interpretation of medical images.

The models we are dealing with below are meant to contribute to the extraction of information about cardiac performance from cardiac images. Most of the models we present are meant to facilitate segmentation, but this is not always the case. The model we propose in this paper is meant to improve the segmentation stage. We focus on methods that are applicable to short axis cardiac MRI. The most thorough survey of such methods to date was done by Frangi et al. (2001). The authors classified all models as *surface models*, *volume models* or *deformation models* and we follow this classification in this brief exposition.

Surface models focus on modeling the epicardial and/or endocardial wall. Such models usually deform by position-

ing the boundaries of the model at the locations of the strongest local features such as edges. In (Pentland et al., 1991), the authors recover a set of global descriptors of motion, which they use to model an approximately elastic object. They use the so called *governing equation*

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{R}$$

where \mathbf{U} is a vector of the model's surface nodes' displacements over time and \mathbf{M} , \mathbf{C} , \mathbf{K} are matrices that model the object's mass, damping and material stiffness. \mathbf{R} is a vector describing the forces acting on the nodes. Similar work based on modeling the LV as an elastic object was presented by other researchers (McInerney et al., 1995; Cohen et al., 1991; Cohen and Cohen, 1992). These represent some of the first efforts at 3D modeling of the heart through so called *deformable models*. Park et al. (1996) used a deformable superquadric to create a volumetric model of the LV and RV of the heart from tagged MRI data. Park and Metaxas introduce parameters for modeling local shape variation. These parameters provide a few intuitive indicators of the heart function, such as the degree of contraction/expansion of the heart. The main advantage of the model is that it is a compact model, as it provides a very small number of parameters that accurately describe the cardiac function. Bardinet et al. (1996) simultaneously segment and track a dynamic sequence of 3D images during a complete cardiac cycle. They accomplish this using a combination of superquadrics-based deformations and *free form deformations*. There has also been a significant amount of research on ASM-based models for cardiac segmentation (van Assen et al., 2006; Davatzikos et al., 2003; Cootes and Taylor, 1995; Shen and Davatzikos, 2000, 2001; de Bruijne et al., 2003; Nain et al., 2006; Tölle et al., 2006). Rousson et al. (2004) have presented a methodology for incorporating more knowledge about cardiac structure through the use of level set methods and ASMs, and could potentially offer a robust approach to cardiac MRI segmentation. In (Delhay et al., 2005), the authors present a 3D surface model which also considers a temporal constraint to increase the accuracy of the segmentation.

In contrast to the vast number of surface models which admittedly constitute a sizeable part of the literature on 3D cardiac models, there have been fewer volume models. Volume models focus on modeling the entire cardiac volume, and not just the surface as surface models do. One such model was introduced in (Mitchell et al., 2002). The authors apply the work of Cootes and Taylor (2004) on statistical point distribution models (PDMs), to model cardiac shape variation. The main advantage of this work is the ease with which shape deformations can be learned from a given training set, and the model's ability to also incorporate the appearance variation of the model. Such models have been extremely successful in the medical imaging field, leading to the creation of many commercially successful products (Cootes and Taylor, 2004). Numerous similar approaches have appeared since then. For example (Kaus et al., 2004), demonstrate the usefulness

of integrating prior sources of knowledge into a model for robust segmentation. Sermesant et al. (2003) present a model which simulates the electromechanical properties of the heart in a computationally efficient way. By incorporating *a priori* knowledge about cardiac properties, the fitting/segmentation of cardiac images should become more robust. In (Lorenzo-Valdés et al., 2004), the authors use the EM algorithm with a probabilistic 4D atlas of the heart for automatic atlas-based segmentation. The EM algorithm is used to estimate initial model parameters and integrate *a priori* information in the model. Promising results are presented.

Deformation models rely on explicit correspondences between tissue landmarks across time frames. They model the tissue deformation over time, incorporating expert knowledge of the temporal deformation of the heart. Hamarneh and Gustavsson (2004), present a 3D ASM of the heart, where the third dimension corresponds to time, and models the temporal deformation of an object across time. The authors present some examples of the method's applicability for cardiac segmentation from ultrasound images. Amini and Duncan (1992) introduce a model in which they estimate the cardiac wall motion with the help of a metric measuring the curvature properties of the surface. Benayoun et al. (1998) use an adaptive mesh to estimate the non-rigid motion of sequences of 3D images. Meshes at two time frames are adjusted in such a way that image differential properties, such as curvature and gradient, are matched. Another recently introduced model is described in Papademetris et al. (2002) and Papademetris (2000), in which the authors propose a deformation model based on principles from continuum mechanics. They employ these properties to calculate the strain throughout the heart wall, which in turn, is used in combination with surface curvature properties to calculate the displacement of the cardiac wall during the cardiac cycle. Work on these ideas continued and was further expanded by researchers at Yale. See for example Yan et al. (2005). We continue with presenting a methodology for fitting 3D AAMs, providing an initial segmentation of our MRI data.

2. Methods

2.1. Optimization of 3D AAMs for short axis cardiac MRI segmentation

Active appearance models (AAMs) provide a robust approach for the analysis of medical images (Cootes and Taylor, 1998, 2004). The ability of AAMs to learn the 3D structure of the heart and not lead to unlikely segmentations has stirred up interest in the medical imaging community regarding their use for the segmentation of the left ventricle from short axis cardiac MRI (Frangi et al., 2001; Mitchell et al., 2002).

The algorithms described in the literature for fitting AAMs, are either efficient but ad hoc algorithms or more robust but inefficient gradient descent type algorithms.

Standard optimization methods for fitting AAMs are inefficient, due to the high number of parameters needing to be optimized. This problem is exacerbated with 3D AAMs since such models often use 50–100 parameters to model the appearance and shape of the object. To deal with this, efficient algorithms for fitting AAMs have been developed (Cootes and Taylor, 1998). Under such formulations, we look for a constant matrix \mathbf{R} such that if the current fitting error between the AAM and the image is $\delta \mathbf{t}$, the update to the AAM parameters is $\delta \mathbf{p} = \mathbf{R} \delta \mathbf{t}$. However, the fitting accuracy and convergence rates of such algorithms are often unsatisfactory. Such methods lack a sound theoretical basis since in general there is no reason why the error measure $\delta \mathbf{t}$ should uniquely identify the update $\delta \mathbf{p}$ of the parameters.

A novel algorithm for the efficient fitting of 2D AAMs was recently introduced in (Matthews and Baker, 2004). Its applicability was demonstrated for face tracking on artificial and real world data. However, as it is cited in (Matthews and Baker, 2004), there was no known way of extending the algorithm to higher dimensions – for fitting 3D or 4D AAMs for example – since an argument used in the paper applied only to 2D similarity transformations. In this section, we present an extension of the algorithm for the fitting of 3D AAMs on short axis cardiac MRI and we demonstrate that the resulting segmentation is at least as accurate as Gauss–Newton-based optimization – one of the most widely used algorithms for solving such problems. By definition, short axis cardiac MR images are such that the long axis of the heart is perpendicular to the image's acquisition plane – even though in practice, due to various sources of error, the long axis is never exactly perpendicular to the image plane. Therefore, during the AAM fitting we only need to rotate our model around the long axis of the heart. We take advantage of this fact to design an efficient and accurate fitting algorithm for 3D AAMs. We use this algorithm to extract from our MRI sequence a preliminary segmentation. We observe border positioning errors that are significantly smaller than the errors reported for other 3D AAMs (Mitchell et al., 2002) which use the constant matrix approach for the fitting. Section 2.2 discusses various ways of further improving this segmentation.

Fig. 1a presents an example of a short axis cardiac MR image. A stack of such images provides a volumetric representation of the heart. Manual segmentations of the left ventricle provide contours that can be used to represent the endocardium and epicardium of the left ventricle. By uniformly sampling each of these contours at i_0 points along their arclength, starting from the closest point to the posterior junction of the left and right ventricles, we can represent each contour by the same number of landmarks. As shown in Fig. 1b, by stacking the landmarks on top of each other, we obtain a 3D representation of the left ventricle's endocardium and epicardium. However, the number of images intersecting the left ventricle is not identical for every patient. Therefore, we need to interpo-

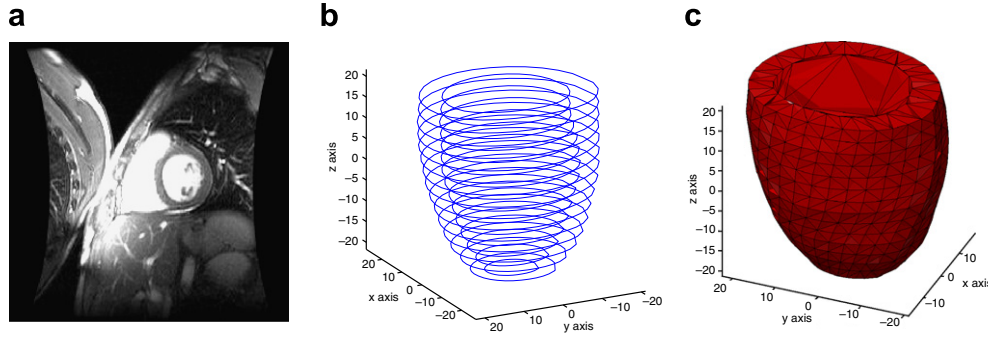


Fig. 1. (a) Short axis cardiac MRI. (b) Endocardial and epicardial landmarks stacked on top of each other. Shown as curves for greater clarity. The mean of these landmarks is given by vector \mathbf{s}_0 . (c) Tetrahedrization of \mathbf{s}_0 . Each tetrahedron represents a part of the myocardial muscle or a part of the left ventricle's blood pool.

late between the contours so that every 3D model is made up of the same number of slices. If we want to create a slice at height z_0 located between two slices, we can simply do the following: From the line segment joining the i th landmark in the two slices, we determine the location with height z_0 . This gives us the i th landmark in the new slice. In our implementation, we have created 15 contour slices, which are evenly sampled along the z -axis, located between the apex and basal contours of the heart.

Let N denote the number of shape samples. If each sample consists of m landmarks, we can represent each shape sample as a $3m$ -dimensional vector. By applying principal component analysis (PCA) on the distribution of the shape vectors, we can approximate any shape \mathbf{s} out of the N shapes as

$$\mathbf{s} \approx \mathbf{s}_0 + \sum_{i=1}^n p_i \mathbf{s}_i \quad (1)$$

for some $\mathbf{p} = (p_1, \dots, p_n) \in \mathbb{R}^n$, where \mathbf{s}_0 is the mean shape vector (also referred to as the *base mesh* in the literature), and \mathbf{s}_i indicates the i th eigenvector. Usually in the literature, authors sum over n eigenvectors \mathbf{s}_i such that the model explains around 90–95% of the shape variation.

When we are building a 2D AAM, we must make sure that the shapes are aligned with each other so that any difference between two shapes due to a similarity transform is removed, or at least minimized. See Cootes and Taylor (2004) for more details. This leads to more compact AAMs, that are described by a smaller number of basis vectors. In (Matthews and Baker, 2004) and in our algorithm, this is a necessary step. By removing the similarity transform from the set of training shapes, the \mathbf{s}_i vectors are orthogonal to a subspace of the 3D similarity transforms of \mathbf{s}_0 , and we use this orthogonality property to achieve efficient fitting of 3D AAMs. We aligned the shapes by using an iterative alignment algorithm as described in Cootes and Taylor (2004). However, in this case we aligned the shapes with respect to translation, scaling and rotation around only the z -axis. We did not align the shapes with respect to x - and y -axis rotations as we only wanted our model to handle rotations around the z -axis.

We need to model the appearance variation of the shape. We first manually split \mathbf{s}_0 in tetrahedra, as shown in Fig. 1c. This splits the left ventricular volume enclosed by \mathbf{s}_0 into tetrahedra. The same landmark connectivity defining the tetrahedra of \mathbf{s}_0 can be used to define the tetrahedrization of any shape variation resulting from Eq. (1). Then, we use these tetrahedra to sample the appearance enclosed by each training shape (Mitchell et al., 2002). Let the mean appearance we get by averaging the sampled appearances be $A_0(\mathbf{x})$ and the k eigenvectors we found by PCA, describing the appearance variation, be $A_1(\mathbf{x}), A_2(\mathbf{x}), \dots, A_k(\mathbf{x})$ (where \mathbf{x} denotes the appearance coordinate in the base model \mathbf{s}_0 coordinate system). For different b_i values

$$A(\mathbf{x}) = A_0(\mathbf{x}) + \sum_{i=1}^k b_i A_i(\mathbf{x}) \quad (2)$$

defines the appearance variations the model has learned from our training data.

2.1.1. The inverse compositional algorithm

There is a wealth of literature on image alignment algorithms and the reader is referred to Baker et al. (2004), Baker and Matthews (2001), Maintz and Viergever (1998), Zitova and Flusser (2003) and the references therein for an overview. The inverse compositional approach can provide fast and reliable image alignment.

Assume we have a template $A_0(\mathbf{x})$ that we want to align with an image $I(\mathbf{x})$. In the compositional framework for image alignment we compute a warp $\mathbf{W}(\mathbf{x}; \Delta \mathbf{p})$ that is composed with the current warp $\mathbf{W}(\mathbf{x}; \mathbf{p})$ (where \mathbf{p} are warp parameters and \mathbf{x} denotes pixel/voxel coordinates), in order to find the update parameters $\Delta \mathbf{p}$ minimizing

$$\sum_{\mathbf{x} \in \text{Domain}(A_0)} [I(\mathbf{W}(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p}); \mathbf{p})) - A_0(\mathbf{x})]^2 \quad (3)$$

where $\text{Domain}(A_0)$ consists of the pixels/voxels \mathbf{x} inside the tetrahedrization of \mathbf{s}_0 (Fig. 1c). In the inverse compositional approach, we are trying to minimize

$$\sum_{\mathbf{x} \in \text{Domain}(A_0(\mathbf{W}))} [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - A_0(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p}))]^2 \quad (4)$$

where $\text{Domain}(A_0(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p})))$ consists of the points \mathbf{x} such that $\mathbf{W}(\mathbf{x}; \Delta \mathbf{p})$ is a point in the domain of A_0 . The solution to this least squares problem is approximately (Baker et al., 2004):

$$\Delta \mathbf{p} = \mathbf{H}^{-1} \sum_{\mathbf{x} \in \text{Domain}(A_0)} \left[\nabla A_0 \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - A_0(\mathbf{x})] \quad (5)$$

where

$$\mathbf{H} = \sum_{\mathbf{x} \in \text{Domain}(A_0)} \left[\nabla A_0 \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[\nabla A_0 \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right] \quad (6)$$

and $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is evaluated at $(\mathbf{x}; 0)$. It can be shown that within first order, under the condition of a fairly “smooth” warp function \mathbf{W} , Eq. (4) is equal to

$$\sum_{\mathbf{x} \in \text{Domain}(A_0)} [I(\mathbf{W}(\mathbf{W}^{-1}(\mathbf{x}; \Delta \mathbf{p}); \mathbf{p})) - A_0(\mathbf{x})]^2 \quad (7)$$

This means that the $\Delta \mathbf{p}$ in Eq. (5) can also be used to minimize Eq. (7). Notice that Eq. (7) is an image alignment error measure of the form Eq. (3). Once we have found $\Delta \mathbf{p}$, we can update the warp by

$$\mathbf{W}(\mathbf{x}; \mathbf{p}) \leftarrow \mathbf{W}(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}^{-1}(\mathbf{x}; \Delta \mathbf{p}) \quad (8)$$

and go to Eq. (5) to perform another iteration of the image alignment algorithm. This is a very efficient algorithm, known as the inverse compositional algorithm (Baker and Matthews, 2001). It is efficient because we can precompute $\nabla A_0 \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ and \mathbf{H} . This algorithm can easily be used for fitting templates $A(\mathbf{x})$ as described above. In that case, if in Eq. (4) above, $A_0(\mathbf{x})$ is replaced by $A(\mathbf{x})$, with appearance variation as described in Eq. (2), then within first order the error is equal to (Baker and Matthews, 2001; Matthews and Baker, 2004):

$$\sum_{\mathbf{x} \in \text{Domain}(A_0)} [\text{proj}_{\text{span}(A_i)^\perp} (I(\mathbf{W}(\mathbf{W}^{-1}(\mathbf{x}; \Delta \mathbf{p}); \mathbf{p})) - A_0(\mathbf{x}))]^2 \quad (9)$$

where $\text{span}(A_i)^\perp$ denotes the space orthogonal to A_1, \dots, A_k . We minimize expression (9) by using Eqs. (5) and (6), only that now we project each column of $\nabla A_0 \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ onto $\text{span}(A_i)^\perp$ and use this projected matrix instead of $\nabla A_0 \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$.

2.1.2. Inverse compositional fitting of 3D AAMs

In the next three subsections, we show how to efficiently fit 3D AAMs using the inverse compositional algorithm. In Section 2.1.3, we show how to extend (Matthews and Baker, 2004) to fit 3D AAMs with no global similarity transformations. In Section 2.1.4, we extend the work and from Section 2.1.3, we show how to fit a 3D AAM when we allow translations, rotations around only one axis – by convention we rotate around the z -axis – and scaling of the coordinate axis. Finally in Section 2.1.5, we put everything together and discuss how we can efficiently update the parameters during each iteration of the algorithm. We end up with the algorithm that we use to fit 3D AAMs.

2.1.3. Fitting without global shape transform

We now show how the inverse compositional algorithm can be used for fitting 3D AAMs without taking care of any global shape similarity transformations (translation, scalings and rotations).

We first need to define $\mathbf{W}(\mathbf{x}; \mathbf{p})$, where \mathbf{p} denotes the current landmarks model parameters from Eq. (1), and the $\mathbf{x} = (x, y, z)^\top$ parameter denotes a point in the base mesh \mathbf{s}_0 . Then $\mathbf{W}(\mathbf{x}; \mathbf{p})$ denotes the warping of \mathbf{x} under the current warp parameters \mathbf{p} . As mentioned above, every base mesh voxel \mathbf{x} lies in a tetrahedron \mathbf{T}_0 defined by the vertices (x_i^0, y_i^0, z_i^0) , (x_j^0, y_j^0, z_j^0) , (x_k^0, y_k^0, z_k^0) , (x_l^0, y_l^0, z_l^0) . If the current shape parameters of the model are \mathbf{p} , then let the vertices of the deformed tetrahedron \mathbf{T}_1 be (x_i, y_i, z_i) , (x_j, y_j, z_j) , (x_k, y_k, z_k) , (x_l, y_l, z_l) which were computed from Eq. (1). $\mathbf{W}(\mathbf{x}; \mathbf{p})$ computes the affine transformation of \mathbf{x} from \mathbf{T}_0 to \mathbf{T}_1 . If $\alpha_i, \alpha_j, \alpha_k, \alpha_l$ denote the barycentric coordinates of \mathbf{x} in \mathbf{T}_0 given by

$$\begin{pmatrix} \alpha_i \\ \alpha_j \\ \alpha_k \\ \alpha_l \end{pmatrix} = \begin{pmatrix} x_i^0 & x_j^0 & x_k^0 & x_l^0 \\ y_i^0 & y_j^0 & y_k^0 & y_l^0 \\ z_i^0 & z_j^0 & z_k^0 & z_l^0 \\ 1 & 1 & 1 & 1 \end{pmatrix}^{-1} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (10)$$

(by the definition of barycentric coordinates $\alpha_i + \alpha_j + \alpha_k + \alpha_l = 1$ and $0 \leq \alpha_i, \alpha_j, \alpha_k, \alpha_l \leq 1$), the affine transformation of \mathbf{x} is $\mathbf{W}(\mathbf{x}; \mathbf{p})$ and is given by

$$\alpha_i(x_i, y_i, z_i)^\top + \alpha_j(x_j, y_j, z_j)^\top + \alpha_k(x_k, y_k, z_k)^\top + \alpha_l(x_l, y_l, z_l)^\top \quad (11)$$

To compute $I(\mathbf{W}(\mathbf{x}; \mathbf{p}))$ in Eq. (5) we do the following: for every point \mathbf{x} in the mean tetrahedrization \mathbf{s}_0 we compute $\mathbf{W}(\mathbf{x}; \mathbf{p})$ and sample image I at that location by trilinear interpolation. Then

$$\frac{\partial \mathbf{W}}{\partial \mathbf{p}} = \sum_{i=1}^m \left[\frac{\partial \mathbf{W}}{\partial x_i} \frac{\partial x_i}{\partial \mathbf{p}} + \frac{\partial \mathbf{W}}{\partial y_i} \frac{\partial y_i}{\partial \mathbf{p}} + \frac{\partial \mathbf{W}}{\partial z_i} \frac{\partial z_i}{\partial \mathbf{p}} \right] \quad (12)$$

where

$$\frac{\partial \mathbf{W}}{\partial x_i} = (\alpha_i, 0, 0)^\top \pi(\mathbf{x}, i) \quad (13)$$

$$\frac{\partial \mathbf{W}}{\partial y_i} = (0, \alpha_i, 0)^\top \pi(\mathbf{x}, i) \quad (14)$$

$$\frac{\partial \mathbf{W}}{\partial z_i} = (0, 0, \alpha_i)^\top \pi(\mathbf{x}, i) \quad (15)$$

and

$$\frac{\partial x_i}{\partial \mathbf{p}} = (\mathbf{s}_1^{x_i}, \mathbf{s}_2^{x_i}, \dots, \mathbf{s}_n^{x_i}) \quad (16)$$

$$\frac{\partial y_i}{\partial \mathbf{p}} = (\mathbf{s}_1^{y_i}, \mathbf{s}_2^{y_i}, \dots, \mathbf{s}_n^{y_i}) \quad (17)$$

$$\frac{\partial z_i}{\partial \mathbf{p}} = (\mathbf{s}_1^{z_i}, \mathbf{s}_2^{z_i}, \dots, \mathbf{s}_n^{z_i}) \quad (18)$$

$\pi(\mathbf{x}, i)$ equals 1 if \mathbf{x} is in a tetrahedron of \mathbf{s}_0 having landmark i as its vertex, and is 0 otherwise. $\mathbf{s}_j^{x_i}, \mathbf{s}_j^{y_i}, \mathbf{s}_j^{z_i}$ denote

the element of \mathbf{s}_j corresponding to x_i , y_i and z_i , respectively. The only nonzero summation terms in Eq. (12), are the terms corresponding to the four vertices of the tetrahedron enclosing the point \mathbf{x} where we are evaluating the Jacobian.

By the argument in Matthews and Baker (2004), within first order $\mathbf{W}^{-1}(\mathbf{x}; \Delta \mathbf{p}) = \mathbf{W}(\mathbf{x}; -\Delta \mathbf{p})$. From Eq. (8), we conclude that we need to find a parameter \mathbf{p}' such that $\mathbf{W}(\mathbf{x}; \mathbf{p}') = \mathbf{W}(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}^{-1}(\mathbf{x}; \Delta \mathbf{p})$. We can approximate this quantity by finding a \mathbf{p}'' such that $\mathbf{W}(\mathbf{x}; \mathbf{p}'') = \mathbf{W}(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}(\mathbf{x}; -\Delta \mathbf{p})$. The problem is that piecewise affine warping does not form a group under the operation of composition since the composition of two piecewise affine warps cannot necessarily be described by another piecewise affine warp. We compensate for this by estimating a new position for the landmarks \mathbf{s}_0 under the composition of the two warps. Once we have done this for all landmarks in \mathbf{s}_0 , we estimate \mathbf{p}'' by finding the closest vector \mathbf{p} in Eq. (1) satisfying the new landmarks and letting $\mathbf{p}'' = \mathbf{p}$.

We estimate a new position for landmarks \mathbf{s}_0 by the following method. For every landmark \mathbf{a} in vector \mathbf{s}_0 , we estimate $\mathbf{W}(\mathbf{a}; -\Delta \mathbf{p})$ by using Eq. (1) with $-\Delta \mathbf{p}$ as parameter. Then, to compute $\mathbf{W}(\mathbf{W}(\mathbf{a}; -\Delta \mathbf{p}); \mathbf{p})$, we use the following heuristic procedure. For each one of the tetrahedra in \mathbf{s}_0 having \mathbf{a} as a vertex, we estimate the destination of $\mathbf{W}(\mathbf{a}; -\Delta \mathbf{p})$ under that tetrahedron's affine warp and we define the value of $\mathbf{W}(\mathbf{W}(\mathbf{a}; -\Delta \mathbf{p}); \mathbf{p})$ to be the average value of the destination of $\mathbf{W}(\mathbf{a}; -\Delta \mathbf{p})$ under the affine warps of those tetrahedra.

2.1.4. Fitting without x - and y -axis rotations

Let $\mathbf{x} = (x, y, z)^T$ and $\mathbf{q} = (q_1, q_2, q_3, q_4, q_5, q_6) = \left(\frac{a}{c_1}, \frac{b}{c_2}, \frac{c}{c_3}, \frac{t_x}{c_4}, \frac{t_y}{c_5}, \frac{t_z}{c_6}\right)$, where the c_i are constants which are defined later. Assume $\mathbf{N}(\mathbf{x}; \mathbf{q})$ is a global transform, which does not rotate the shape around the x - and y -axis, defined as follows:

$$\mathbf{N}(\mathbf{x}; \mathbf{q}) = \begin{pmatrix} 1+a & -b & 0 \\ b & 1+a & 0 \\ 0 & 0 & 1+c \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (19)$$

Notice that $\mathbf{q} = 0$ gives the identity transformation. If we let $a = k \cos(\theta) - 1$, $b = k \sin(\theta)$ and $c = s - 1$ then

$$\mathbf{N}(\mathbf{x}; \mathbf{q}) = \begin{pmatrix} k \cos(\theta) & -k \sin(\theta) & 0 \\ k \sin(\theta) & k \cos(\theta) & 0 \\ 0 & 0 & s \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (20)$$

This performs a rotation by an angle θ around the z -axis followed by a scaling by k of the x, y coordinates and a scaling by s of the z -coordinates, followed by a translation by $(t_x, t_y, t_z)^T$. If we replace the s above by k , then we are performing a three-dimensional similarity transform where we are not rotating around the x -, y -axis. In other words Eq. (19) above has slightly more expressive power than a typical similarity transform since it allows to scale the

z -coordinate values by a value different from the scaling of the x, y coordinates. Then $\mathbf{N} \circ \mathbf{W}$ performs both the piecewise affine warp of Section 2.1.3 and the global transform \mathbf{N} .

One can argue against scaling along the z -axis independently from the x -, y -axis scalings, since this is not a similarity transform. But from the test cases we performed later on, this does not lead to incorrect model instances. On the contrary, it adds more expressive power to our model, making it easier to fit heart models with different z -axis scalings than the ones in our training set.

As noted above, our base mesh is $\mathbf{s}_0 = (x_1^0, y_1^0, z_1^0, \dots, x_m^0, y_m^0, z_m^0)^T$. Let

$$\begin{aligned} \mathbf{s}_1^* &= c_1(x_1^0, y_1^0, 0, \dots, x_m^0, y_m^0, 0)^T \\ \mathbf{s}_2^* &= c_2(-y_1^0, x_1^0, 0, \dots, -y_m^0, x_m^0, 0)^T \\ \mathbf{s}_3^* &= c_3(0, 0, z_1^0, \dots, 0, 0, z_m^0)^T \\ \mathbf{s}_4^* &= c_4(1, 0, 0, \dots, 1, 0, 0)^T \\ \mathbf{s}_5^* &= c_5(0, 1, 0, \dots, 0, 1, 0)^T \\ \mathbf{s}_6^* &= c_6(0, 0, 1, \dots, 0, 0, 1)^T \end{aligned}$$

where c_i is a constant such that \mathbf{s}_i^* is of unit length. Then

$$\mathbf{N}(\mathbf{s}_0; \mathbf{q}) = \mathbf{s}_0 + \sum_{i=1}^6 q_i \mathbf{s}_i^* \quad (21)$$

where $q_1 = \frac{a}{c_1}$, $q_2 = \frac{b}{c_2}$, $q_3 = \frac{c}{c_3}$, $q_4 = \frac{t_x}{c_4}$, $q_5 = \frac{t_y}{c_5}$, $q_6 = \frac{t_z}{c_6}$. If during the shape alignment we aligned the training data such that their center was at point $(0, 0, 0)$ then $S^* = \{\mathbf{s}_1^*, \mathbf{s}_2^*, \mathbf{s}_3^*, \mathbf{s}_4^*, \mathbf{s}_5^*, \mathbf{s}_6^*\}$ is an orthonormal set.

The set $S = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n\}$ from Eq. (1) is the set of eigenvectors of the covariance matrix. For reasons which will become obvious later, we must make sure that every vector in S^* is orthogonal to every vector in S . Because S^* is not a similarity transform, and due to various sources of error, the alignment procedure we performed earlier might not make the two sets fully orthogonal. We therefore have to orthogonalize the two sets S^* and S . In our test cases we did it as follows. For every vector in $\mathbf{v} \in S$ we take its projection \mathbf{v}' onto the space orthogonal to S^* by

$$\mathbf{v}' = \mathbf{v} - \sum_{\mathbf{v}^* \in S^*} (\mathbf{v}^T \mathbf{v}^*) \mathbf{v}^* \quad (22)$$

We call this new set S' . Then, by using an orthogonalization algorithm such as the Gram–Schmidt algorithm we transform S' into S'' , where now S'' is an orthonormal set. Then every vector in S^* is orthogonal to every vector in S'' .

We need to show what is the Jacobian of $\mathbf{N} \circ \mathbf{W}$ which is used instead of $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ in Eqs. (5) and (6), and how to update the parameters from Eq. (7). We see that the Jacobian of $\mathbf{N} \circ \mathbf{W}$ is $\left(\frac{\partial \mathbf{N} \circ \mathbf{W}}{\partial \mathbf{q}}, \frac{\partial \mathbf{N} \circ \mathbf{W}}{\partial \mathbf{p}}\right) = \left(\frac{\partial \mathbf{N}}{\partial \mathbf{q}}, \frac{\partial \mathbf{W}}{\partial \mathbf{p}}\right)$ since we are evaluating the Jacobian at $\mathbf{p} = 0$ and $\mathbf{q} = 0$, which implies that $\mathbf{W}(\mathbf{x}; \mathbf{p}) = \mathbf{N}(\mathbf{x}; \mathbf{q}) = \mathbf{x}$.

2.1.5. Updating the parameters

As noted in Section 2.1.3, to within first order $(\mathbf{N} \circ \mathbf{W})^{-1}(\mathbf{x}; \Delta \mathbf{q}, \Delta \mathbf{p}) = \mathbf{N} \circ \mathbf{W}(\mathbf{x}; -\Delta \mathbf{q}, -\Delta \mathbf{p})$. We use this approximation to define

$$\mathbf{s}^\dagger = \mathbf{N} \circ \mathbf{W}((\mathbf{N} \circ \mathbf{W})^{-1}(\mathbf{s}_0; \Delta \mathbf{q}, \Delta \mathbf{p}); \mathbf{q}, \mathbf{p}) \quad (23)$$

(the new locations of the landmarks \mathbf{s}_0) by using the method of Section 2.1.3 for composing two warps. Once we have estimated the new landmark positions \mathbf{s}^\dagger from Eq. (23), we need to find new values for \mathbf{p} and \mathbf{q} such that $\mathbf{N} \circ \mathbf{W}(\mathbf{s}_0; \mathbf{q}, \mathbf{p}) = \mathbf{s}^\dagger$. First notice that

$$\mathbf{N} \circ \mathbf{W}(\mathbf{s}_0; \mathbf{q}, \mathbf{p}) = \mathbf{N} \left(\mathbf{s}_0 + \sum_{i=1}^n p_i \mathbf{s}_i; \mathbf{q} \right) \quad (24)$$

which can be rewritten as

$$\mathbf{N}(\mathbf{s}_0; \mathbf{q}) + \begin{pmatrix} 1+a & -b & 0 \\ b & 1+a & 0 \\ 0 & 0 & 1+c \end{pmatrix} \sum_{i=1}^n p_i \mathbf{s}_i \quad (25)$$

where the summations above are taking place over all vectors $\mathbf{s}_i \in S''$. The matrix multiplication in Eq. (25) above with the $3m$ -dimensional vector \mathbf{s}_i , indicates the result of multiplying every triple of adjacent x, y, z coordinates by the matrix.

By using Eq. (25), we see that $\mathbf{N} \circ \mathbf{W}(\mathbf{s}_0; \mathbf{q}, \mathbf{p}) = \mathbf{s}^\dagger$ can be rewritten as

$$\begin{aligned} \mathbf{s}^\dagger = & \mathbf{s}_0 + \sum_{i=1}^6 q_i \mathbf{s}_i^* + (1+a) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \sum_{i=1}^n p_i \mathbf{s}_i \\ & + b \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \sum_{i=1}^n p_i \mathbf{s}_i + (1+c) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \sum_{i=1}^n p_i \mathbf{s}_i \end{aligned} \quad (26)$$

The three terms in Eq. (26) above that are multiplied by $1+a$, b and $1+c$, are orthogonal to the vectors in S^* , since S^* and S'' are orthogonal. This is most difficult to see for the fourth term in Eq. (26) that is multiplied by b . This term is orthogonal to the vectors in S^* because for each vector in S^* , if we switch the values of the x and y coordinates and change the sign of one of the two coordinates, the resulting vector still belongs to $\text{span}(S^*)$. From Eq. (26), we get

$$q_i = \mathbf{s}_i^* \cdot (\mathbf{s}^\dagger - \mathbf{s}_0) \quad (27)$$

which we can use to get

$$p_i = \mathbf{s}_i \cdot (\mathbf{N}^{-1}(\mathbf{s}^\dagger; \mathbf{q}) - \mathbf{s}_0) \quad (28)$$

where

$$\mathbf{N}^{-1}(\mathbf{s}^\dagger; \mathbf{q}) = \begin{pmatrix} 1+a & -b & 0 \\ b & 1+a & 0 \\ 0 & 0 & 1+c \end{pmatrix}^{-1} \left[\mathbf{s}^\dagger - \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \right] \quad (29)$$

and we find the new parameters \mathbf{p} and \mathbf{q} . Without the orthonormal set S^* we introduced, there would be no efficient way of updating the parameters \mathbf{p}, \mathbf{q} .

This algorithm can also be applied to 3D AAMs, the third dimension of which corresponds to time. A straightforward extension of this algorithm is possible for 4D AAMs, at least in theory, which simultaneously encode the model variations along the x, y, z , and temporal dimension, assuming that we are only interested in rotating the x and y coordinates of the model. The most significant change to our algorithm would entail redefining our orthonormal basis S^* . In this case if $\mathbf{s}_0 = (x_1^0, y_1^0, z_1^0, t_1^0, \dots, x_l^0, y_l^0, z_l^0, t_l^0)^T$ and the center of gravity of the points in the training set is $(0, 0, 0, 0)$, we can let

$$\mathbf{s}_1^* = c_1 (x_1^0, y_1^0, 0, 0, \dots, x_l^0, y_l^0, 0, 0)^T \quad (30)$$

$$\mathbf{s}_2^* = c_2 (-y_1^0, x_1^0, 0, 0, \dots, -y_l^0, x_l^0, 0, 0)^T \quad (31)$$

$$\mathbf{s}_3^* = c_3 (0, 0, z_1^0, t_1^0, \dots, 0, 0, z_l^0, t_l^0)^T \quad (32)$$

$$\mathbf{s}_4^* = c_4 (1, 0, 0, 0, \dots, 1, 0, 0, 0)^T \quad (33)$$

$$\mathbf{s}_5^* = c_5 (0, 1, 0, 0, \dots, 0, 1, 0, 0)^T \quad (34)$$

$$\mathbf{s}_6^* = c_6 (0, 0, 1, 0, \dots, 0, 0, 1, 0)^T \quad (35)$$

$$\mathbf{s}_7^* = c_7 (0, 0, 0, 1, \dots, 0, 0, 0, 1)^T \quad (36)$$

where c_i is a constant such that \mathbf{s}_i^* is of unit length. Then S^* is an orthonormal basis which causes a rotation around the z and temporal axis (it rotates the x, y coordinates only). Rigorous testing would need to be performed to see if such a model would work in practice. The high dimensionality could potentially cause the inverse compositional algorithm not to converge. This is a topic for future research.

2.2. Hierarchical 2D + time ASMs

We introduce a hierarchical 2D + time ASM whose modes of deformation depend on shapes' wavelet transform coefficients (Davatzikos et al., 2003). We demonstrate that this leads to ASMs which generalize better in the presence of small training sets, and we use this approach to integrate temporal constraints in the segmentation process and to further improve the segmentation acquired using the AAM from Section 2.1.

2.2.1. Wavelets and active shape models

As we have already indicated, AAMs and ASMs need a training set in order to learn their allowable modes of deformation. In many ways, this is both an advantage and a disadvantage.

It is an advantage because it makes the model simple, since it no longer depends on complicated physical models to constrain its allowable deformations. Models which encode elasticity or other smoothness constraints do not necessarily reflect true biological variability. Moreover, extensive usage of such models has shown that they can be too flexible, making them vulnerable to getting trapped in spurious image structures, thus, offering bad solutions. This makes the manual initialization of such elastic models a necessity. Since AAMs and ASMs learn from a training set of allowable deformations, we have greater guarantees

that the deformations of the model reflects the true variability of the object. This removes many of the drawbacks with elastic type models.

However, the reliance on a training set, is also a disadvantage, because it implies that the expressive power of the model is constrained by the size of its training set. The number of eigenvectors that can be used to represent a shape, is bounded by the number of training samples. In many research areas such as medical imaging, it is quite difficult to obtain a large enough training set so that the model can learn the true shape variability. In our cardiac MRI case for example, consider the case where we are confronted with an image which is an outlier, such as an image of a severely diseased and abnormal heart. Then, it is likely that our model does not have enough modes of deformation to deform sufficiently and fit the image, if a similar example is not present in the training set.

The higher the degree of variability of our object, the greater this problem becomes. The training size needed rises dramatically for 3D and 4D statistical shape models. Indeed, this is one of the reasons why there has been little research in the use of 3D and 4D cardiac models using statistical models of appearance. It can be argued that novel methods for making such models more expressively powerful in the presence of small training sets are needed to build a clinically validated statistical model for cardiac segmentation.

Various solutions have been proposed for handling the problem of artificially enlarging small training sets (Davatzikos et al., 2003; Cootes and Taylor, 1995; Shen and Davatzikos, 2000, 2001; de Bruijne et al., 2003; Nain et al., 2006; Tölli et al., 2006). The only work we are aware of comparing such approaches is (Lötjönen et al., 2005). One of the most common suggestions is to use so called *hierarchical active shape model*, in which the surface of the shape is split into patches and PCA is applied on each one of these patches. This means that we are now applying PCA on vectors of lower dimensionality, and, thus, the number of training examples relative to the number of landmarks increases. However, this can sometimes lead to the problem of losing any knowledge of the connectivity of the shape patches and their deformation in unison.

Recently, Davatzikos et al. (2003) presented a hierarchical active shape model that was based on the wavelet transform and a multiresolution representation of deformable contours. It generalizes the idea of learning the model deformation at different scales, by using wavelets (Mallat, 1999), a more elegant way of creating hierarchical ASMs. The authors tested its applicability for the segmentation of the corpus callosum in magnetic resonance images and for the segmentation of hand contours from images. They observed obvious improvements in the segmentation accuracy compared to a standard ASM when the models were given a small training set, and some smaller improvements when tested against standard hierarchical ASMs. This motivated us to investigate the potential applications of the extension of this method to

3D ASMs, since as we have previously discussed, 3D Statistical models tend to have generalization problems when confronted with outliers.

In (Davatzikos et al., 2003), the authors use such a wavelet transformation of the spectrum of the shape to learn the typical variations of the wavelet coefficients on different scales, and control the modes of deformation of the ASM by applying PCA on various bands of the decomposition of the spectrum. They show that such a model generalizes better and is better able to handle small localized deformations as well as global deformations of the object. We describe below the work that Davatzikos et al. (2003) did for 2D, and then we proceed to describe our 3D extension.

The third dimension of the ASM will correspond to the temporal axis of the shape, rather than to the z -axis that the AAM encoded. In this way, we are also encoding knowledge about the temporal deformation of the left ventricle. We use this model to improve the segmentation from the previous section.

2.2.2. Wavelet decomposition-based hierarchical 2D ASMs

Assume we have N two-dimensional training shapes, where each training shape is made up of K landmarks. For simplicity, we assume that each training shape is represented by a closed contour. Assume that the training shapes have already been aligned using Procrustes analysis (Cootes and Taylor, 2004). Let $\mathbf{u}_n = (u_{n1}, u_{n2}, \dots, u_{nK})^T$ be a vector representing the x -coordinates of the n th training shape, and $\mathbf{v}_n = (v_{n1}, v_{n2}, \dots, v_{nK})^T$ be a vector representing the y -coordinates of the n th shape. By applying a wavelet transformation $\mathbf{W}\mathbf{D}$ on \mathbf{u}_n and \mathbf{v}_n we get their wavelet transformation coefficients concatenated into a single vector

$$\mathbf{c}_n = \mathbf{W}\mathbf{D}(\mathbf{u}_n) = (c_{n1}, c_{n2}, \dots, c_{nK})^T \quad (37)$$

$$\mathbf{d}_n = \mathbf{W}\mathbf{D}(\mathbf{v}_n) = (d_{n1}, d_{n2}, \dots, d_{nK})^T \quad (38)$$

The authors in (Davatzikos et al., 2003) use a wavelet transform to decompose the time–frequency plane of each one of these 1D signals \mathbf{u}_n , \mathbf{v}_n into the same number of bands. Then, in order to evaluate the typical modes of deformation corresponding to that band, they apply PCA on the coefficients belonging to each one of those bands. In other words, let $\mathbf{e}_n = [\mathbf{c}_n^T, \mathbf{d}_n^T]^T$ and $\mathbf{w}_n^{(b)}$ be a vector containing the wavelet coefficients of \mathbf{e}_n corresponding to the b th band. So $\mathbf{w}_n^{(b)}$ contains wavelet coefficients from both \mathbf{c}_n and \mathbf{d}_n . By performing PCA on the set $\{\mathbf{w}_n^{(b)}\}_{n=1, \dots, N}$, containing the wavelet coefficients of the b th bands in our data set, the model learns the typical values of the b th band's coefficients. Determining the bands into which we wish to split the time–frequency plane is crucial. By splitting it into too few bands, we can end up with negligible gains in expressive power, while splitting it into too many bands can lead to a model which over generalizes and leads to unlikely shape instances, thus leading to bad segmentations.

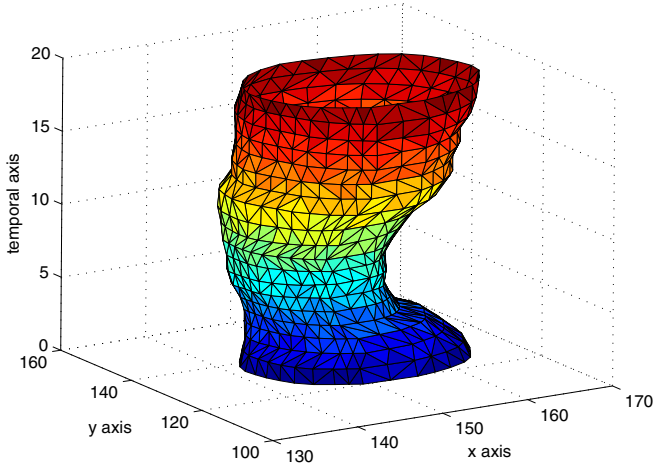


Fig. 2. Example of a possible \mathbf{c}_{endo} function.

2.2.3. Wavelet decomposition-based hierarchical 2D + time ASMs

The main difficulty in extending this approach to 3D is finding a parameterization of the surface of the model, so that any point on its surface can be given by two parametric coordinates (s, t) . We define two parametric functions to accomplish this, one for the endocardial surface (\mathbf{c}_{endo}) and one for the epicardial surface of the left ventricle (\mathbf{c}_{epi}), where $\mathbf{c}_{\text{endo}} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ and $\mathbf{c}_{\text{epi}} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ (see Fig. 2).

Let $\mathbf{c}_{\text{endo}}(s, t) = (u_{\text{endo}}(s, t), v_{\text{endo}}(s, t), t)$ and $\mathbf{c}_{\text{epi}}(s, t) = (u_{\text{epi}}(s, t), v_{\text{epi}}(s, t), t)$. The t parameter represents the frame number and the s parameter is the arclength parameterization of the current contour. The u_{endo} and u_{epi} parameterize the x coordinates of our model, and the v_{endo} , v_{epi} parameterize its y coordinates. For each frame t , the model is made up of two almost concentric contours representing the endocardium and epicardium of the left ventricle. Since the cardiac cycle is periodic, we can view \mathbf{c}_{endo} and \mathbf{c}_{epi} as periodic in t . Furthermore, since both the endocardium and epicardium are closed contours we can also view \mathbf{c}_{endo} and \mathbf{c}_{epi} as periodic in s . This greatly simplifies the wavelet analysis, since if the contours were not periodic, many of the high magnitude wavelet coefficients would have occurred due to signal discontinuities, and not due to some other important deformation of the model. However, we should mention that in practice, MRI data is not fully periodic across time and in practice the data might contain discontinuities, especially towards the end of the sequence. Despite this, and as we show in our experiments, the above mentioned model provides excellent results.

We perform a 2D wavelet decomposition up to level 2, on each one of $u_{\text{endo}}(s, t)$, $u_{\text{epi}}(s, t)$, $v_{\text{endo}}(s, t)$, $v_{\text{epi}}(s, t)$. In other words, we create four wavelet trees of height 2 (Mallat, 1999), where at each depth of the tree we recursively apply a single level 2D wavelet transform to its approximation coefficients, as shown in Fig. 3. The coefficients corresponding to each leaf of a 2D wavelet tree are

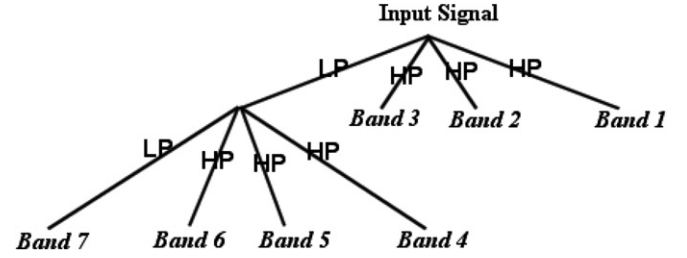


Fig. 3. Example of the 2D wavelet tree we used, as well as the bands in which we grouped the wavelet coefficients. HP denotes the application of a filter offering us detail coefficients, and LP corresponds to the application of a filter providing approximation coefficients. Notice that to each band we assign the wavelet coefficients corresponding to a leaf of the tree.

assigned to a different band,¹ for a total of seven bands. Each of the endocardial and epicardial contours is made up of 32 landmarks and there are 20 time frames in each patient's sequence, which implies that we can view each of $u_{\text{endo}}(s, t)$, $u_{\text{epi}}(s, t)$, $v_{\text{endo}}(s, t)$, $v_{\text{epi}}(s, t)$ as a 32×20 matrix. We could resample the contours to end up with more sampling points, but for the sake of simplicity we do not do so. Assume that we have N such training shapes on which we have applied the above wavelet decomposition.

Let us assume that $\mathbf{c}_{\text{endo}}^{(b)}$, $\mathbf{c}_{\text{epi}}^{(b)}$, $\mathbf{d}_{\text{endo}}^{(b)}$, $\mathbf{d}_{\text{epi}}^{(b)}$ are vectors containing the wavelet coefficients of the b th band of $u_{\text{endo}}(s, t)$, $u_{\text{epi}}(s, t)$, $v_{\text{endo}}(s, t)$, $v_{\text{epi}}(s, t)$, respectively.

Let $\mathbf{e}_n^{(b)} = [\mathbf{c}_{\text{endo}}^{(b)}, \mathbf{c}_{\text{epi}}^{(b)}, \mathbf{d}_{\text{endo}}^{(b)}, \mathbf{d}_{\text{epi}}^{(b)}]$ be a vector corresponding to the b th band of the n th training shape (so we assume that $u_{\text{endo}}(s, t)$, $u_{\text{epi}}(s, t)$, $v_{\text{endo}}(s, t)$, $v_{\text{epi}}(s, t)$ parameterize the n th training shape). Then, we use PCA on the set $\{\mathbf{e}_n^{(b)}\}_{n=1, \dots, N}$ and retain 90% of the variation to create a model/eigenvector basis $\Phi^{(b)}$ of the modes of deformation of the coefficients of the b th band. Let $\mu^{(b)}$ be the mean vector of the wavelet coefficients $\{\mathbf{e}_n^{(b)}\}_{n=1, \dots, N}$. Finally, let \mathbf{s} represent the landmarks of a shape. The algorithm by which we deform \mathbf{s} until it segments the image is the following:

1. Using scalings, rotations around the temporal axis and x -, y -axis translations, align shape \mathbf{s} with the mean shape \mathbf{s}_0 of our training set. Call this aligned shape $T(\mathbf{s})$.
2. Apply a level 2 wavelet decomposition to $T(\mathbf{s})$, as described in this section, to acquire its wavelet transform coefficients. Let $\mathbf{s}^{(b)}$ be a vector containing the wavelet coefficients of the b th band.
3. Let $\mathbf{s}_{\text{new}}^{(b)} = \mu^{(b)} + \Phi^{(b)} \mathbf{p}^{(b)}$ where $\mathbf{p}^{(b)} = (\Phi^{(b)})^T (\mathbf{s}^{(b)} - \mu^{(b)})$. If an entry of $\mathbf{p}^{(b)}$ is greater than two standard deviations ($2 * \text{SD}$) of its typical value, set that entry of $\mathbf{p}^{(b)}$ to $2 * \text{SD}$.
4. Use the inverse discrete wavelet transform to reconstruct the shape from the $\mathbf{s}_{\text{new}}^{(b)}$ vectors. Call this reconstructed shape \mathbf{s}_{new} .

¹ For simplicity, we follow the same terminology used by Davatzikos et al. (2003), by referring to the sets into which we group the wavelet coefficients, as the *bands* of the decomposition.

5. If a convergence criterion is satisfied we are successful, and $T^{-1}(\mathbf{s}_{\text{new}})$ denotes the segmented shape, where T^{-1} is the inverse of T . Otherwise, use the methodology described below to move the landmarks of $T^{-1}(\mathbf{s}_{\text{new}})$ to more salient positions on the image, let $\mathbf{s} \leftarrow T^{-1}(\mathbf{s}_{\text{new}})$, and go to step 2.

The methodology used to move the landmarks of $T^{-1}(\mathbf{s}_{\text{new}})$ to more salient positions on the image is based on a slightly modified version of the Cootes and Taylor (2004) algorithm for fitting ASMs, that searches exclusively for edges and not lines. For each landmark, the algorithm searches along the normal direction \mathbf{n} of the landmark's position on a contour, searching for the strongest edge position. In other words, for each of the candidate landmark positions \mathbf{l}_p , we sample the image along a $2k + 1$ pixel length whisker $I_p(i)$ with direction \mathbf{n} , where the landmark \mathbf{l}_p is positioned at whisker index $i = k + 1$. The strongest edge position for an endocardial landmark is at the position \mathbf{l}_p where $\sum_{i=1}^k (I_p(i) - I_p(2k + 1 - i))$ is maximized (assuming lower indices correspond to image regions closer to the center of the left ventricle, and the left ventricle's blood pool is encoded with higher intensity values than the myocardium). Similarly, the strongest edge position for an epicardial landmark is defined as the landmark position \mathbf{l}_p where $\sum_{i=1}^k |I_p(i) - I_p(2k + 1 - i)|$ is maximized. Notice that we use the absolute value for the epicardial landmarks due to the wide variability in image intensity outside of the epicardial contour – the endocardial region of the right ventricle usually has a much higher intensity value than the other regions outside the left ventricle's epicardium. In other words, this landmark search algorithm can be seen as a version of the Cootes and Taylor (2004) algorithm for fitting ASMs, only that it searches exclusively for edges (not lines). Since the 3D AAM provided a very good initialization of the segmentation, we limit our search along three pixels in direction \mathbf{n} and three pixels in direction $-\mathbf{n}$. The main advantage of this approach is that no training phase is needed, and the algorithm is less sensitive to image intensity changes across subjects.

We use the above algorithm to improve the 3D AAM segmentation of Section 2.1. The initial shape \mathbf{s} that we use during the first iteration of the above algorithm, is the segmentation we got using the 3D AAM. In the next section, we describe our testing methodology and our results, which prove that this approach leads to significant improvements in LV segmentation.

3. Experimental setup

3.1. Test data

The dataset on which we trained and fitted the models was comprised of short axis cardiac MR image sequences acquired from 33 subjects, for a total of 7980 2D images which were provided by the Department of Diagnostic Imaging of the Hospital for Sick Children in Toronto,

Canada. The images were scanned with a GE Genesis Signa MR scanner using the FIESTA scan protocol. Most of the subjects displayed a variety of heart abnormalities such as cardiomyopathy, aortic regurgitation, enlarged ventricles and ischemia, and a smaller number displayed a left ventricle related abnormality, providing a realistic and challenging performance benchmark. All the subjects were under the age of 18. Each patient's image sequence consisted of exactly 20 frames and the number of slices acquired along the long axis of the subjects ranged between 8 and 15. Spacing-between slices ranged between 6 and 13 mm. Each image slice consisted of 256×256 pixels with a pixel-spacing of 0.93–1.64 mm. Each image where both the endocardial and epicardial contours of the left ventricle (Rosse and Gaddum-Rosse, 1997) were visible, was manually segmented, to provide the ground truth. The endocardial contours were drawn behind the papillary muscles and trabeculae. Each contour was drawn starting from the junction of the left and right ventricles located closest to the posterior interventricular sulcus. The manual segmentation was performed by the first author and took approximately 3 weeks of full time work. This resulted in 5011 manually segmented MR images, with a total of 10,022 endocardial and epicardial contours, on which we have tested our approach. We tested our algorithm on all the frames of the cardiac cycle. This is important as the segmentation becomes much more challenging the closer we get to the end-systolic frame of a sequence. All test cases were done on an Intel Xeon 3.06 Ghz with 3 GB RAM, using Matlab 2006a. The dataset of cardiac MR images and their manual segmentations are available online at <http://www.cse.yorku.ca/~mridataset/> and at the journal's website as an Electronic Annex.

3.2. Test protocol

3.2.1. Model building and segmentation

We performed a model initialization at the beginning of the 3D AAM fitting stage. A similarity transformation for the mean model \mathbf{s}_0 was estimated, that aligned \mathbf{s}_0 with manual delineations of the most basal, the median and the most apical slices at end-diastole of the raw MRI sequence. Each frame's resulting segmentation by the 3D AAM was used to initialize the model in the next frame. The fitting process stopped if the error change was below a certain threshold, or if the fitting error had stopped decreasing monotonically. We used three-fold cross-validation to segment the data: We split the data set into three groups, each group consisting of 11 subjects. From each one of these three groups we built two 3D appearance models. One model used the data from 10 frames close to end-diastole and the other model used the data from other 10 frames close to end-systole. Therefore, each of the two models was built using the data from 110 frames. Each group's models were used to segment the respective frames of the other 22 patients' data. As a result, each patient's frame was segmented by two models, by using one model built from each

group of 11 patients. We selected as the final segmentation of each frame the 3D model that minimized the fitting criterion – the appearance error. We refer to this model as IC3 in our results. Please also note that we refer to the selection of the best of two segmentations as the best-of-two approach. We compared the accuracy of IC3 to a standard Gauss–Newton-based optimization approach which simultaneously optimized all the model parameters and used the best-of-two models approach – like IC3 did – to select the final segmentation. We compared our algorithm with a Gauss–Newton optimizer for two main reasons: Firstly, to emphasize the gain in speed that our algorithm can provide compared to classical optimization approaches. Secondly, to investigate our algorithm’s fitting accuracy against an exhaustively tested algorithm which is known from previous research (Baker et al., 2004) to outperform the faster but less reliable approaches described in Section 1. We also compared IC3 to the classical approach of fitting AAMs (Cootes and Taylor, 1998; Mitchell et al., 2002) with independent shape and appearance updating, using a constant matrix \mathbf{R} to determine the parameter updates, as discussed in Section 2.1. Variants of this approach were previously used to optimize similar type of problems with encouraging results (Cootes and Taylor, 2004; Stegmann, 2004). We also experimented with another inverse compositional-based approach to 3D image segmentation, which we refer to as IC9. We used this model to improve the segmentation of the 10 frames closest to end-systole. The difference between IC9 and IC3 is that to segment the 10 frames close to end-systole we used six cardiac models and used a best-of-six approach. The first two models were the ones used to segment the frame in IC3. The other two models were identical but had a different mean shape – we used the same mean shape as was used to segment the diastole data. Finally, the last two models were the ones used to segment the corresponding diastole frames. As a result, each of the 10 volumes close to end-systole was segmented by six models. The model with the best fit – in terms of matching the image intensity – was selected as our final solution. This gave us an indication of how the integration of multiple models affects our segmentation, with very encouraging results as we show in the next section.

For each model, we evenly sampled the voxels comprising the domain of A_0 (see Section 2.1), resulting in roughly 45,000 voxel samples for each model that was built. This corresponded to roughly a one voxel resolution in all image sequences. Once the AAM had converged, for each of the image slices intersected by our model, we extracted the endocardial and epicardial contours of the model’s intersection with the image. These contours represented the segmentation of the respective image. Each model was made up of 30 shape parameters, which accounted for 92% to 94% of the shape variation across the different models’ datasets. Furthermore, each model used 20 appearance parameters – that the inverse compositional algorithm projected out –, which accounted for 76–82% of all appearance

variation after applying the intensity normalization described in Section 3.2.2. Finally, the inverse compositional-based models used six global shape parameters \mathbf{q} , for a total of 56 parameters. The Gauss–Newton and Classical-based models used five global shape parameters (the z -axis scaling was the same as that of the x -, y -axis) so they used 55 parameters.

Once the 3D AAM-based segmentation was complete, we applied the 2D + time ASM-based algorithm to improve the results acquired using each of the four above mentioned 3D AAM optimization algorithms. Again, we used three-fold cross-validation. We split our dataset into the same three groups that we created when training and testing the 3D AAM algorithm. For each subject whose segmentation we wished to improve, we used as a training set the data from the other two groups. Each training sample from those two groups was acquired from 20 image frames, sampled from the same height, and consisted of the manual segmentations of these 20 images. We used as samples the three heights closest to the base where all 20 images had been manually segmented. This gave us a total of $3 \times 22 = 66$ training samples across our 22 training subjects. We also included in our training set the 3D AAM’s solution for that particular patient. This allowed us to encode in the permissible modes of deformation, the knowledge we had gained from the 3D AAM-based segmentation, and also to initialize the model with virtually the same deformation according to which the 3D AAM had segmented the images. We aligned the 3D training data, so that each training sample was as close as possible to the mean shape with respect to x , y coordinate translations, scalings and rotations around the temporal axis. We then used the 3D AAM-based segmentation to initialize the corresponding 2D + time model.

For each subject, we used the 2D + time ASM to re-segment the three 20-frame samples closest to the basal slice that had been segmented by the 3D AAM. For the reasons discussed in Section 1, we used the 2D + time ASM to segment slices that were close to the base of the heart. Typically, those slices have edges with high contrast and respond the best to edge/feature-based segmentation methods. Preliminary testing using the 2D + time ASM to re-segment more apical slices has not demonstrated any significant segmentation improvements.

During the fitting of the model we did not do a similarity transform on the model and all deformations took place by exclusively modifying the modes of deformation of the hierarchical ASM. Since the 3D AAM from the previous section provided a good initialization for our model, by scaling, translating and rotating the model, we could end up making the model too flexible. It is a topic of future research to investigate whether by including similarity transforms to the deformation of the model, we end up with even better results. We use a Daubechies six wavelet. The greater the support of the wavelet, the smoother the reconstructed shapes are (Davatzikos et al., 2003), so some planning is needed to decide which wavelet is more

appropriate for a particular application. For each of the final 2D images that were segmented using both the 3D AAM, and the 2D + time ASM, we measured the standard deviation of the intensity inside the myocardium, and retained the segmentation with the lowest standard deviation, as this is an indicator of a more uniform texture and thus, most likely, an improved segmentation.

3.2.2. Intensity correction

The inverse compositional-based algorithm was proven sensitive to global intensity changes in the images. We handled this by normalizing the left ventricle's intensity in each image to a common mean and standard deviation before training and fitting our models: Firstly, the sampled voxels falling in $\text{Domain}(A_0)$ were normalized to have a mean intensity value of μ_0 and a standard deviation of σ_0 . This was simple to achieve since we could use the manual segmentation of the left ventricle in the training images to determine the voxels mapped into $\text{Domain}(A_0)$. The linear transformation that achieved this for each training image sequence, was applied globally to all the image pixels in the corresponding training sequence. We then proceeded with training the intensity model of our 3D AAM.

Then, in order to normalize the intensity of an image sequence that we wished to segment, we did the following: The manual delineations at end-diastole that were used to initialize our model were also used to build an estimate of the mean μ and standard deviation σ of the left ventricle's intensity. Based on the manual delineations of the endocardium and epicardium at end-diastole, we randomly sampled a number of points from the endocardium and myocardium such that $c\%$ of the points fell inside the endocardium and $100 - c\%$ of the points fell inside the myocardial region. The value of c was chosen to match the percentage of the texture sampling points in $\text{Domain}(A_0)$ that were inside the endocardium. The mean and standard deviation of these points was used to estimate μ and σ . The intensity normalization of all the frames in an image sequence was achieved by transforming the intensity of each pixel i in every frame, to $\frac{(i-\mu)}{\sigma}\sigma_0 + \mu_0$. In other words, we made the simplifying assumption that for all frames, all the image pixels falling inside the left ventricle were due to a common Gaussian variable with mean μ and variance σ^2 . In practice the method provided us with excellent results. For each subject whose image sequence we wished to segment, we used the end-diastolic frame to build a single linear transformation that was in turn applied to all the other frames. In other words, we did not build a distinct intensity correcting linear transformation for each frame of the image sequence. We have attempted building a distinct intensity correcting linear transformation for each frame using the corresponding manual segmentations, but we did not observe any significant changes in the quality of the resulting segmentations. Future work could involve adding a few parameters to the inverse compositional-based model

that could adjust the model's global intensity during the optimization phase so that this preprocessing phase does not need to take place.

3.2.3. Breath-hold correction

The data sets were acquired during breath-hold. Differences in expiration levels typically lead to shifted cardiac structures. This makes it necessary to include a correction phase during the preprocessing step of our dataset, in order to fix this misalignment of short axis image slices. To correct this problem we used a simple image registration algorithm. We used the manual delineation of the median slice

Table 1
Results after fitting the 3D AAM on the data set

Fitting algorithm and metric	Error	Max	Min	Sig
<i>Inv Comp 9 (22.8 s)</i>				
ENDO + EPI volumetric	1.51 ± 0.49	3.34	0.68	N/A
ENDO volumetric	1.51 ± 0.55	4.03	0.57	N/A
EPI volumetric	1.52 ± 0.52	3.60	0.63	N/A
ENDO + EPI landmark	1.53 ± 1.38	15.97	0	N/A
ENDO landmark	1.49 ± 1.37	14.89	0	N/A
EPI landmark	1.56 ± 1.38	15.97	0	N/A
ENDO volume error	6.79 ± 7.44	50.04	0	N/A
EPI volume error	10.33 ± 9.68	82.98	0.02	N/A
Myocardial volume error	10.21 ± 10.76	87.75	0.01	N/A
<i>Inv Comp 3 (8.6 s)</i>				
ENDO + EPI volumetric	1.62 ± 0.60	3.95	0.68	*** (0)
ENDO volumetric	1.68 ± 0.79	6.35	0.57	*** (0)
EPI volumetric	1.58 ± 0.59	4.60	0.63	*** (0)
ENDO + EPI landmark	1.65 ± 1.52	15.97	0	*** (0)
ENDO landmark	1.68 ± 1.59	14.89	0	*** (0)
EPI landmark	1.63 ± 1.47	15.97	0	*** (0)
ENDO volume error	8.77 ± 10.91	98.97	0.01	*(0.0042)
EPI volume error	11.76 ± 12.90	102.17	0.02	*(0.1488)
Myocardial volume error	12.91 ± 16.67	108.22	0.02	*(0.1006)
<i>Gauss-Newton (1013.4 s)</i>				
ENDO + EPI volumetric	1.67 ± 0.81	7.35	0.59	*** (0)
ENDO volumetric	1.47 ± 0.56	4.25	0.60	*(0.0370)
EPI volumetric	1.78 ± 1.00	8.86	0.55	*** (0)
ENDO + EPI landmark	1.64 ± 1.56	17.09	0	*** (0)
ENDO landmark	1.42 ± 1.34	13.89	0	*** (0)
EPI landmark	1.78 ± 1.68	17.09	0	*** (0)
ENDO volume error	5.94 ± 5.83	38.79	0.01	*(0.2649)
EPI volume error	17.35 ± 16.94	92.59	0.05	*** (0)
Myocardial volume error	14.95 ± 15.75	94.94	0.02	*** (0)
<i>Classical (5.4 s)</i>				
ENDO + EPI volumetric	1.75 ± 0.64	5.68	0.74	*** (0)
ENDO volumetric	1.80 ± 0.66	4.93	0.64	*** (0)
EPI volumetric	1.71 ± 0.72	6.74	0.70	*** (0)
ENDO + EPI landmark	1.76 ± 1.56	18.58	0	*** (0)
ENDO landmark	1.78 ± 1.56	13.80	0	*** (0)
EPI landmark	1.75 ± 1.56	18.58	0	*** (0)
ENDO volume error	8.40 ± 7.67	56.12	0	*** (0)
EPI volume error	12.63 ± 14.13	84.44	0.04	*(0.0737)
Myocardial volume error	14.54 ± 14.67	82.74	0.03	*** (0)

Error: the mean and standard deviation of the respective error. Max/min: the maximum/minimum error. Sig: the statistical significance of the difference between the row's error and the respective error measure of the IC9 algorithm. * $0.01 \leq p < 0.05$, ** $0.001 \leq p < 0.01$, *** $0 \leq p < 0.001$, $\times 0.05 \leq p$. In the first column, we indicate the average fitting time per 3D volume using the corresponding algorithm.

to define a square sampling region which completely enclosed the delineation. We then applied a simple optimizer (Gauss–Newton optimization) to determine the translation that minimized the image intensity difference between the square region of the median slice and the corresponding region of the slice above it. Once the optimal shift was determined, all the slices above the median slice were appropriately shifted, and the process repeated recursively with the next slice. This process was then repeated with the slices below the median slice. Note that the translations were always rounded to the nearest integer so that no sub-pixel resampling was needed. There exists in the lit-

erature some related work for breath-hold correction (Löjtjönen et al., 2004; Stegmann et al., 2005).

3.2.4. Quantitative evaluation and tests of significance

To quantitatively evaluate the performance of the algorithm the following metrics were used: the *landmark errors*, the *volumetric errors*, and the *volume errors*. Each manually segmented contour was evenly sampled along its arclength and we estimated the minimum distance of each landmark to a point on the segmentation provided by the particular algorithm we were evaluating. This error measure is referred to as the *landmark error* and it is measured in

Table 2
Results after using the 2D + time ASM on the data set

Fitting algorithm and metric	Error	Max	Min	Sig1	Sig2	Sig3
<i>Inv Comp 9</i>						
ENDO + EPI volumetric	1.47 ± 0.44	3.07	0.68	*** (0)	N/A	N/A
ENDO volumetric	1.43 ± 0.49	3.63	0.57	*** (0)	N/A	N/A
EPI volumetric	1.51 ± 0.48	3.44	0.65	× (0.0524)	N/A	N/A
ENDO + EPI landmark	1.49 ± 1.34	15.97	0	*** (0)	N/A	N/A
ENDO landmark	1.41 ± 1.31	14.89	0	*** (0)	N/A	N/A
EPI landmark	1.55 ± 1.36	15.97	0	** (0.0076)	N/A	N/A
ENDO volume error	6.10 ± 6.44	47.07	0.02	× (0.1187)	N/A	N/A
EPI volume error	9.82 ± 8.97	79.10	0.02	× (0.7816)	N/A	N/A
Myocardial volume error	9.75 ± 9.53	83.31	0.01	× (0.6914)	N/A	N/A
<i>Inv Comp 3</i>						
ENDO + EPI volumetric	1.55 ± 0.54	3.81	0.70	*** (0)	*** (0)	*** (0)
ENDO volumetric	1.55 ± 0.66	5.87	0.61	*** (0)	*** (0)	*** (0)
EPI volumetric	1.55 ± 0.54	4.35	0.64	*** (0)	*** (0)	*** (0)
ENDO + EPI landmark	1.58 ± 1.46	15.97	0	*** (0)	*** (0)	*** (0)
ENDO landmark	1.55 ± 1.49	14.89	0	*** (0)	*** (0)	*** (0)
EPI landmark	1.60 ± 1.45	15.97	0	*** (0)	*** (0)	*** (0)
ENDO volume error	7.58 ± 9.30	85.57	0.01	* (0.0467)	* (0.0175)	*** (0)
EPI volume error	11.13 ± 12.69	100.30	0.03	× (0.3393)	× (0.4189)	× (0.0770)
Myocardial volume error	12.55 ± 15.50	98.79	0.01	× (0.5687)	* (0.0419)	× (0.1762)
<i>Gauss–Newton</i>						
ENDO + EPI volumetric	1.54 ± 0.71	5.97	0.63	*** (0)	** (0.0020)	*** (0)
ENDO volumetric	1.40 ± 0.50	3.80	0.55	*** (0)	× (0.1103)	** (0.0084)
EPI volumetric	1.62 ± 0.85	7.01	0.55	*** (0)	*** (0.0001)	*** (0)
ENDO + EPI landmark	1.52 ± 1.48	17.09	0	*** (0)	*** (0)	*** (0)
ENDO landmark	1.35 ± 1.30	13.89	0	*** (0)	*** (0)	*** (0)
EPI landmark	1.63 ± 1.58	17.09	0	*** (0)	*** (0.0008)	*** (0)
ENDO volume error	5.76 ± 5.35	32.40	0	× (0.9230)	× (0.5134)	× (0.6127)
EPI volume error	13.68 ± 15.19	78.75	0.01	*** (0)	* (0.0228)	*** (0)
Myocardial volume error	12.06 ± 14.01	89.95	0.04	*** (0)	× (0.3329)	*** (0)
<i>Classical</i>						
ENDO + EPI volumetric	1.64 ± 0.56	4.60	0.74	*** (0)	*** (0)	*** (0)
ENDO volumetric	1.64 ± 0.58	4.37	0.62	*** (0)	*** (0)	*** (0)
EPI volumetric	1.64 ± 0.62	5.38	0.69	*** (0)	*** (0)	*** (0)
ENDO + EPI landmark	1.65 ± 1.49	18.58	0	*** (0)	*** (0)	*** (0)
ENDO landmark	1.63 ± 1.49	13.80	0	*** (0)	*** (0)	*** (0)
EPI landmark	1.67 ± 1.49	18.58	0	*** (0)	*** (0)	*** (0)
ENDO volume error	7.24 ± 6.31	43.22	0.04	* (0.0255)	*** (0)	*** (0)
EPI volume error	11.48 ± 12.26	75.66	0	× (0.2181)	× (0.4102)	* (0.0354)
Myocardial volume error	13.51 ± 12.94	74.62	0	× (0.4504)	*** (0)	*** (0)

Error: the mean and standard deviation of the respective error. Max/min: the maximum/minimum error. Each row represents the results after using the 2D + time ASM to improve the row's 3D AAM-based segmentation. Sig1: the statistical significance between the row's error (after using the 2D + time ASM) and the respective error measure in Table 1 (before using the 2D + time ASM). Sig2: the statistical significance between the row's error (after using the 2D + time ASM) and the respective IC9 error measure in Table 1 (before using the 2D + time ASM). Sig3: the statistical significance between the corresponding row's error in Table 1 (before using the 2D + time ASM) and the respective IC9 error measure in this table (after using the 2D + time ASM). *0.01 ≤ p < 0.05, **0.001 ≤ p < 0.01, ***0 ≤ p < 0.001, ×0.05 ≤ p.

mm. The *volumetric error* is defined with respect to a particular subject's frame as the average landmark error of the segmentations in that frame's set of images. It provides

a measure of how well the set of images in that particular frame were segmented and it is also measured in mm. In other words, while each landmark error is defined with

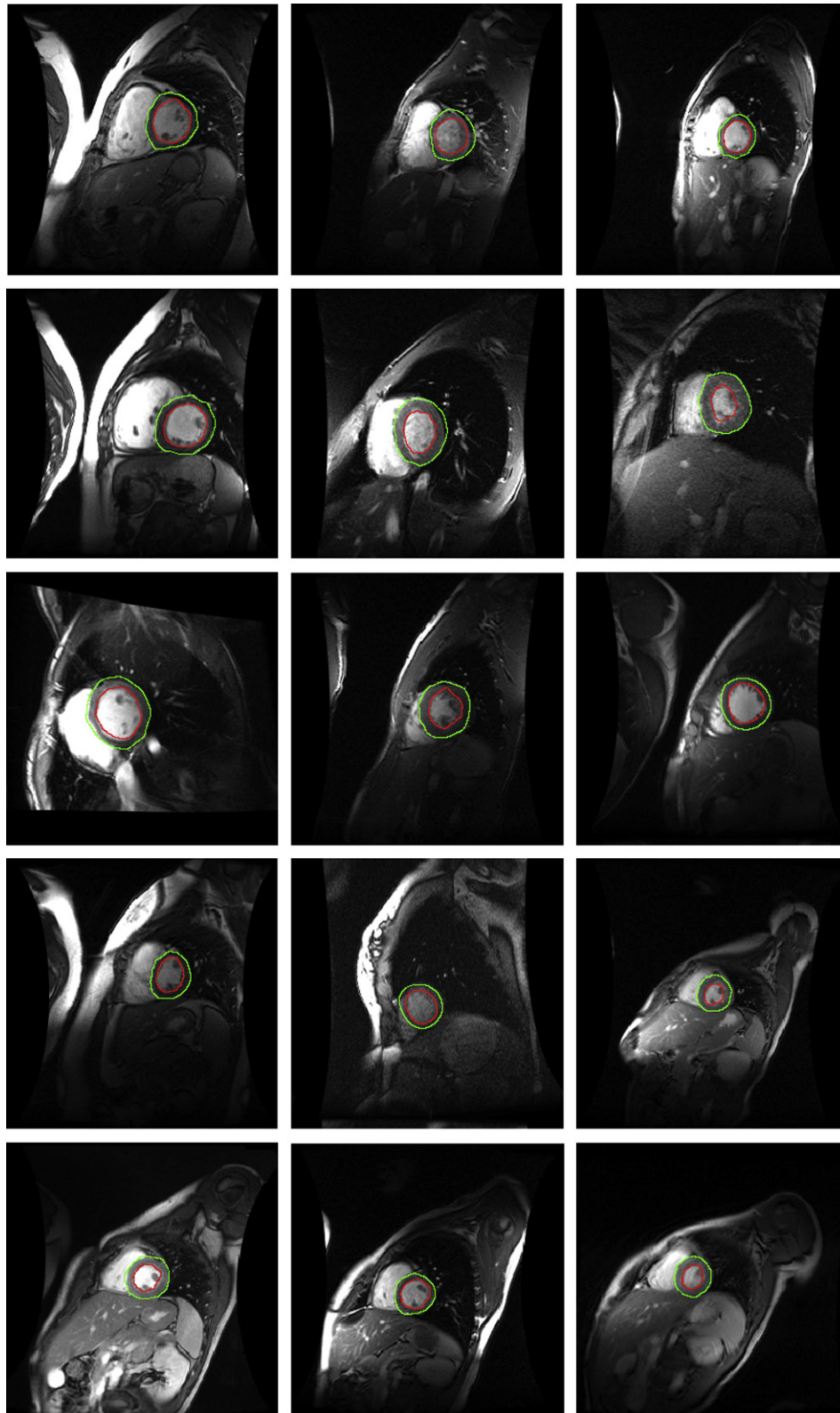


Fig. 4. Resulting segmentations for subjects 1–15 (one random sample from each patient's sequence is displayed) after using IC9 followed by the hierarchical 2D + time-based approach. The endocardial/epicardial segmentation is shown with red/green contours, respectively. (For interpretation of the references to the colour in this figure legend, the reader is referred to the web version of this article.)

respect to a particular sampled landmark on a particular subject's frame, the corresponding volumetric error is the average of all the landmark errors in that subject's frame.

Average error values for the endocardium (ENDO), epicardium (EPI) and both the endocardium and epicardium (ENDO + EPI) are presented. Finally the *volume error* is

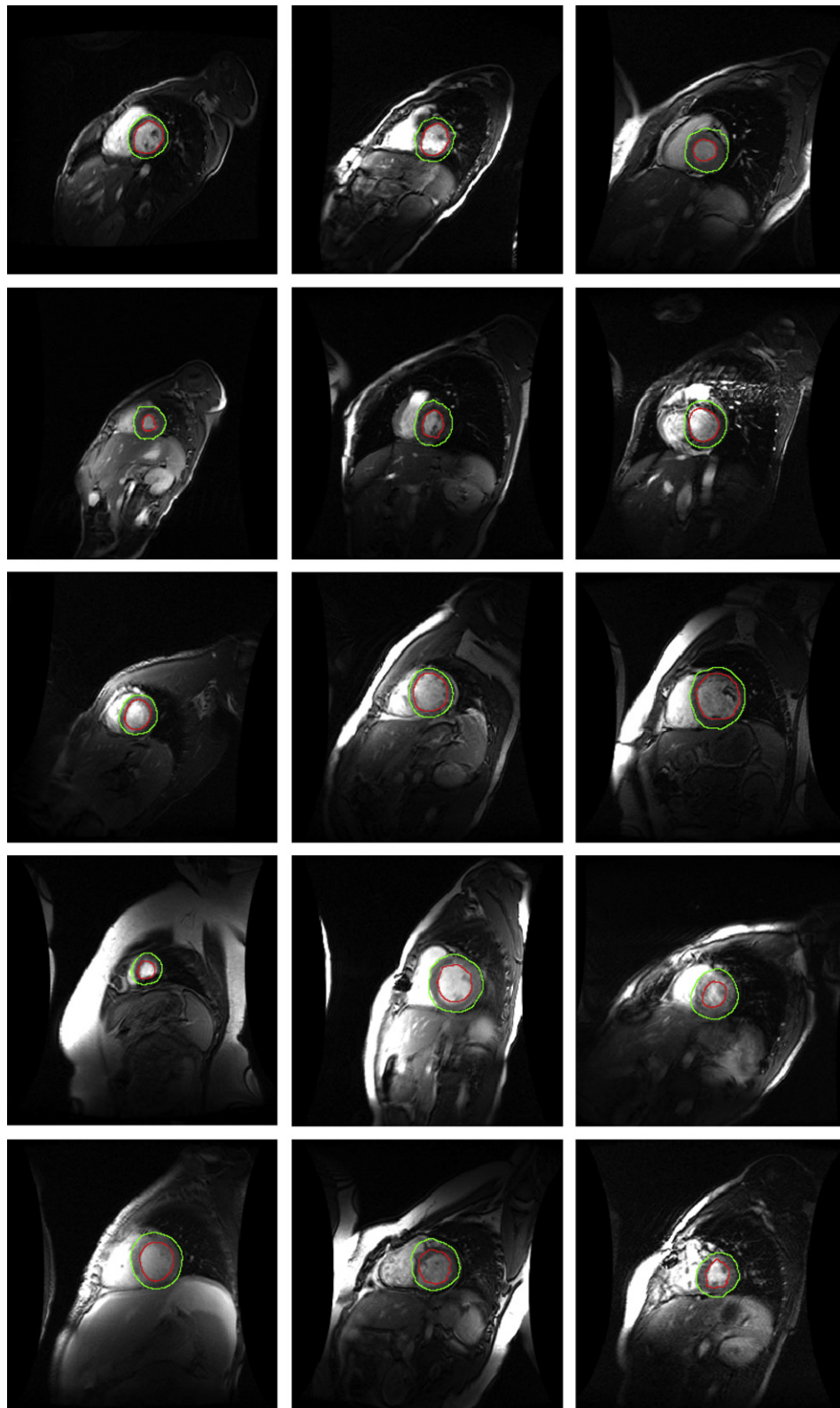


Fig. 5. Resulting segmentations for subjects 16–30 (one random sample from each patient's sequence is displayed) after using IC9 followed by the hierarchical 2D + time-based approach. The endocardial/epicardial segmentation is shown with red/green contours, respectively. (For interpretation of the references to the colour in this figure legend, the reader is referred to the web version of this article.)

measured in cm^3 , and provides a measure of the error (absolute difference) in the volume of various segmented cardiac structures – the volume of the region inside the epicardium (EPI), the endocardial volume (ENDO) and the myocardial volume.

To determine whether differences in the volumetric, landmark and volume errors were significantly different we used *t*-tests and the Wilcoxon test of significance. Error measurements for the volumetric errors were obtained for the same subjects and the error distribution was close to Gaussian, and therefore we used a paired-samples *t*-test. Error measurements for the landmark and volume errors were obtained from the same subjects on image slices that were segmented both manually and by our segmentation algorithms. However, the distribution of the errors was in general non-gaussian and therefore we used the Wilcoxon test of significance for those error measurements.

4. Results

Table 1 presents our results using the four different optimizations approaches for fitting a 3D AAM (IC9, IC3, Gauss–Newton and the Classical approach). The mean time it took to segment each 3D volume is presented in seconds in the table's first column. The error (means \pm standard deviation), maximum error, minimum error and tests of statistical significance of the differences in the errors achieved by different optimization algorithms are presented. Similarly, Table 2 presents the improvements achieved after using the 2D + time ASM to improve the results of Table 1. Note that for IC9, no results are presented for the sig2, sig3 columns since by definition they are equal to the sig1 column. See the tables' captions for more details.

Figs. 4–6 present one random segmentation from each of our dataset's 33 subjects, after applying the 2D + time ASM to improve the results using IC9. Note that each of the displayed images in Figs. 4–6 was selected completely randomly, and thus Figs. 4–6 provide a representative sample of typical segmentations one would observe. The endocardial segmentations are displayed with red contours and the epicardial segmentations are displayed with green contours. Fig. 7 presents the volumetric errors (mean and stan-

dard deviation) of the various 3D AAM optimization algorithms as a function of frame number. In most of our subjects end-systole occurred at frames 8–9 and end-diastole occurred in frame 1. The manual initialization of the model was always performed at frame 1. The end-diastole trained models were applied to frames 1–5, 16–20 and the end-systole trained models were applied to frames 6–15. Fig. 8 shows that our hierarchical 2D + time ASM significantly improves the segmentations – in terms of the mean error, and the error's standard deviation. Comparisons using IC3, IC9, Gauss–Newton and Classical optimization methods before and after applying the 2D + time ASM are presented. Fig. 9 presents regression graphs of the endocardial, epicardial and myocardial volumes estimated using the IC9 algorithm followed by the 2D + time improvement. For each of the endocardial, epicardial, and myocardial volumes the regression is performed on all 20 frames of each subject, on the three frames closest to end-diastole, and on the three frames closest to end-systole. Fig. 10 presents the results after using some popular denoising algorithms to improve the segmentations. We tested a 2D Wiener filter based denoising algorithm and a 2D Wavelet packet-based algorithm.

Table 3 presents previously published results using various popular 3D cardiac segmentation models. Table 4 compares the wavelet-based hierarchical 2D + time ASM with a 2D + time ASM that was built using standard PCA. Results based on training sets of various sizes are presented.

5. Discussion

From inspecting Table 1, we reach the following general conclusions. Overall, based on the ENDO + EPI metrics, IC9 performs the best. It is followed by IC3, then by Gauss–Newton optimization and finally, by the Classical approach. However, in terms of the endocardial (ENDO) volumetric, landmark and volume errors, Gauss–Newton optimization performs the best. However, the difference is not statistically significant compared to IC9 for the ENDO volume error ($p = 0.2649$), and the difference is relatively weak for the ENDO volumetric error ($p = 0.0370$). Overall the best performing algorithm, IC9, runs approximately 44

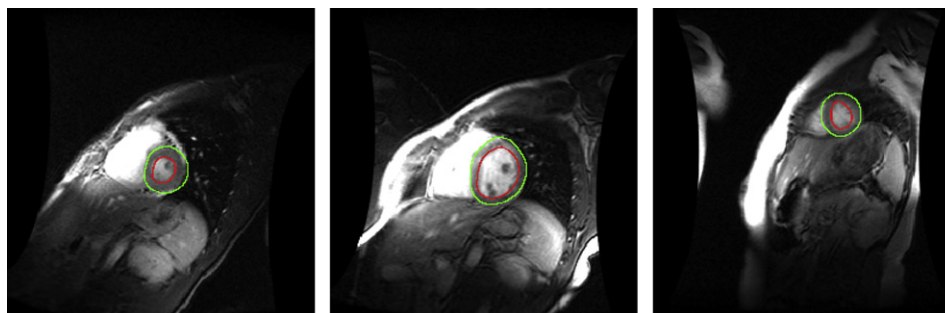


Fig. 6. Resulting segmentations for subjects 31–33 (one random sample from each patient's sequence is displayed) after using IC9 followed by the hierarchical 2D + time-based approach. The endocardial/epicardial segmentation is shown with red/green contours, respectively. (For interpretation of the references to the colour in this figure legend, the reader is referred to the web version of this article.)

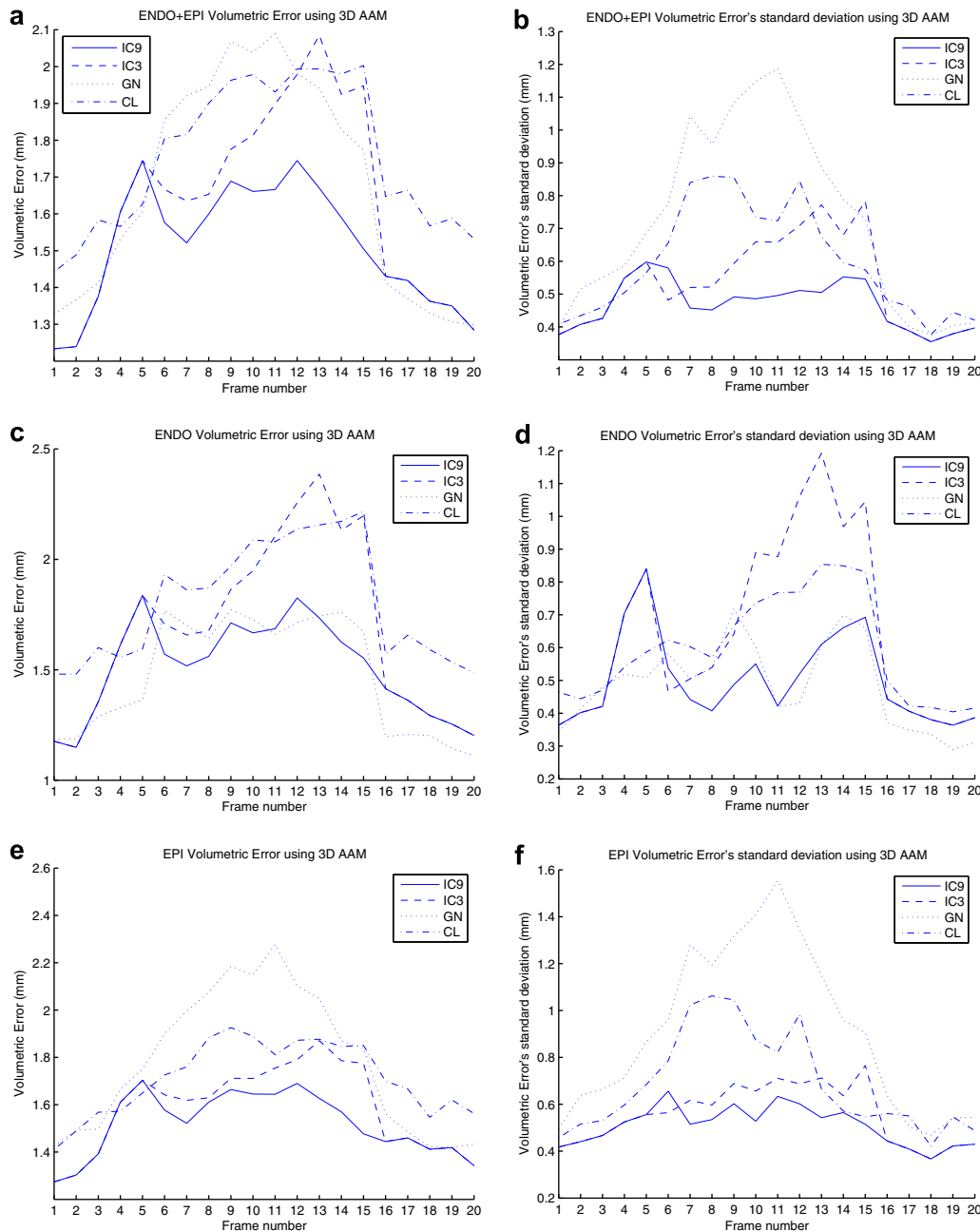


Fig. 7. Mean and standard deviation of the volumetric error using a 3D AAM, as a function of frame number. End-diastole is close to frame 1 and end-systole is close to frames 8–9 in most subjects. IC9/IC3: results after using the respective inverse compositional algorithm; GN: results after using the Gauss–Newton algorithm; CL: results after using the classical formulation; (a and b): results of both endocardial and epicardial segmentations; (c and d): results of endocardial segmentations; (e and f): results of epicardial segmentations.

times faster than standard Gauss–Newton-based optimization and achieves at least comparable results. IC9 and IC3 can be seen as extremely fast and reliable algorithms for segmenting short axis cardiac MRI using 3D AAMs.

We suspect that a reason why Gauss–Newton gives a slightly greater error than IC3 and IC9 is because it also optimizes the appearance parameters, while the inverse compositional-based algorithms project them out. We believe that if both algorithms directly optimized the same number of parameters, Gauss–Newton-based optimization

would be at least as accurate as the inverse compositional algorithm, but would remain much slower.

Fig. 7 demonstrates how the errors evolve as a function of the frame number and how the difficulty of the segmentation problem increases as we get close to end-systole. Note that we used frames 6–15 from our training data to train the systole centered models and the other frames were used to train the diastole centered models. It is interesting to observe that starting from frame 1 and approaching frame 5, when typically the heart has contracted significantly,

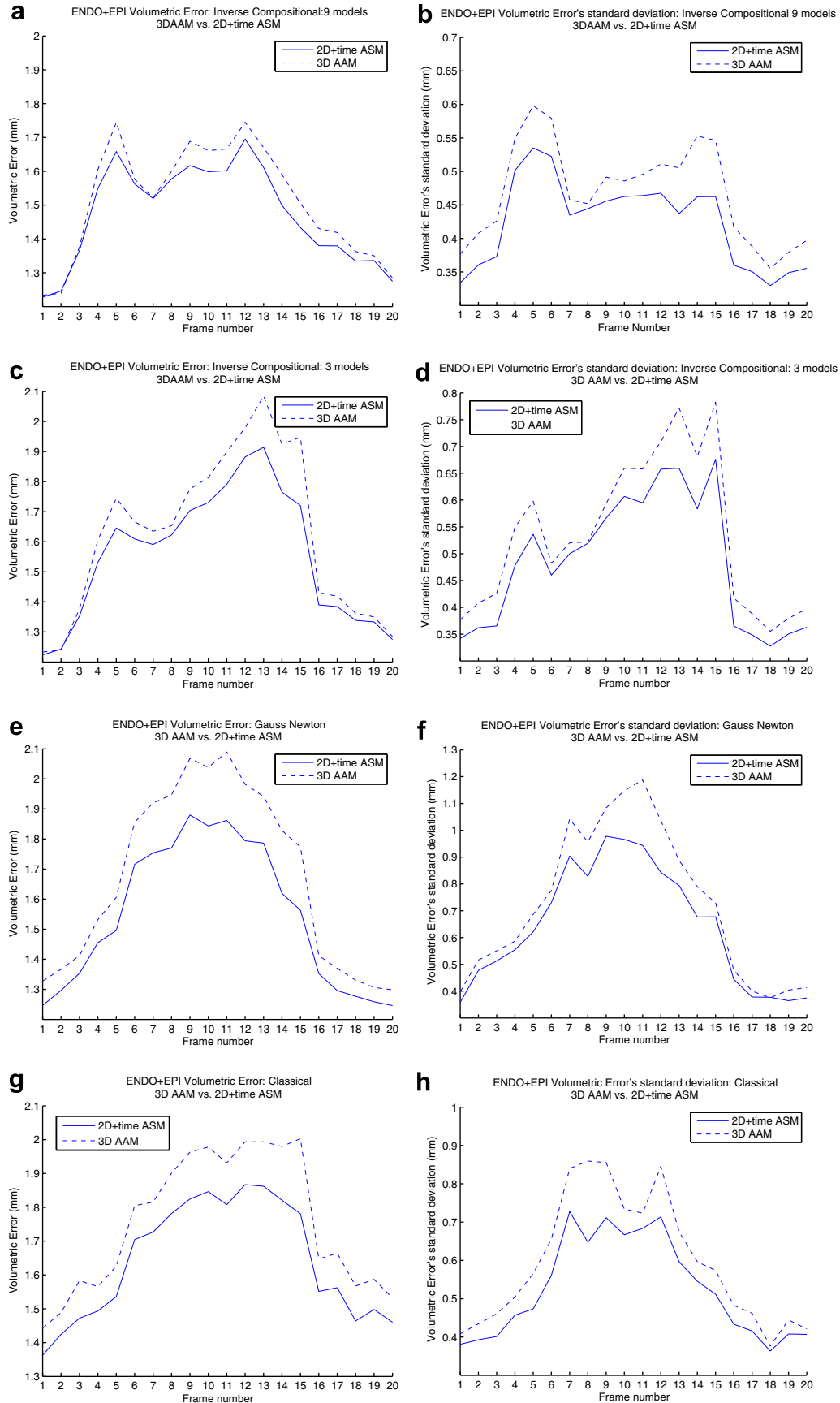


Fig. 8. Mean and standard deviation of the volumetric error, as a function of frame number, before and after using the 2D + time ASM. (a and b): results using IC9; (c and d): results using IC3; (e and f): results using Gauss–Newton; (g and h): results using the classical formulation.

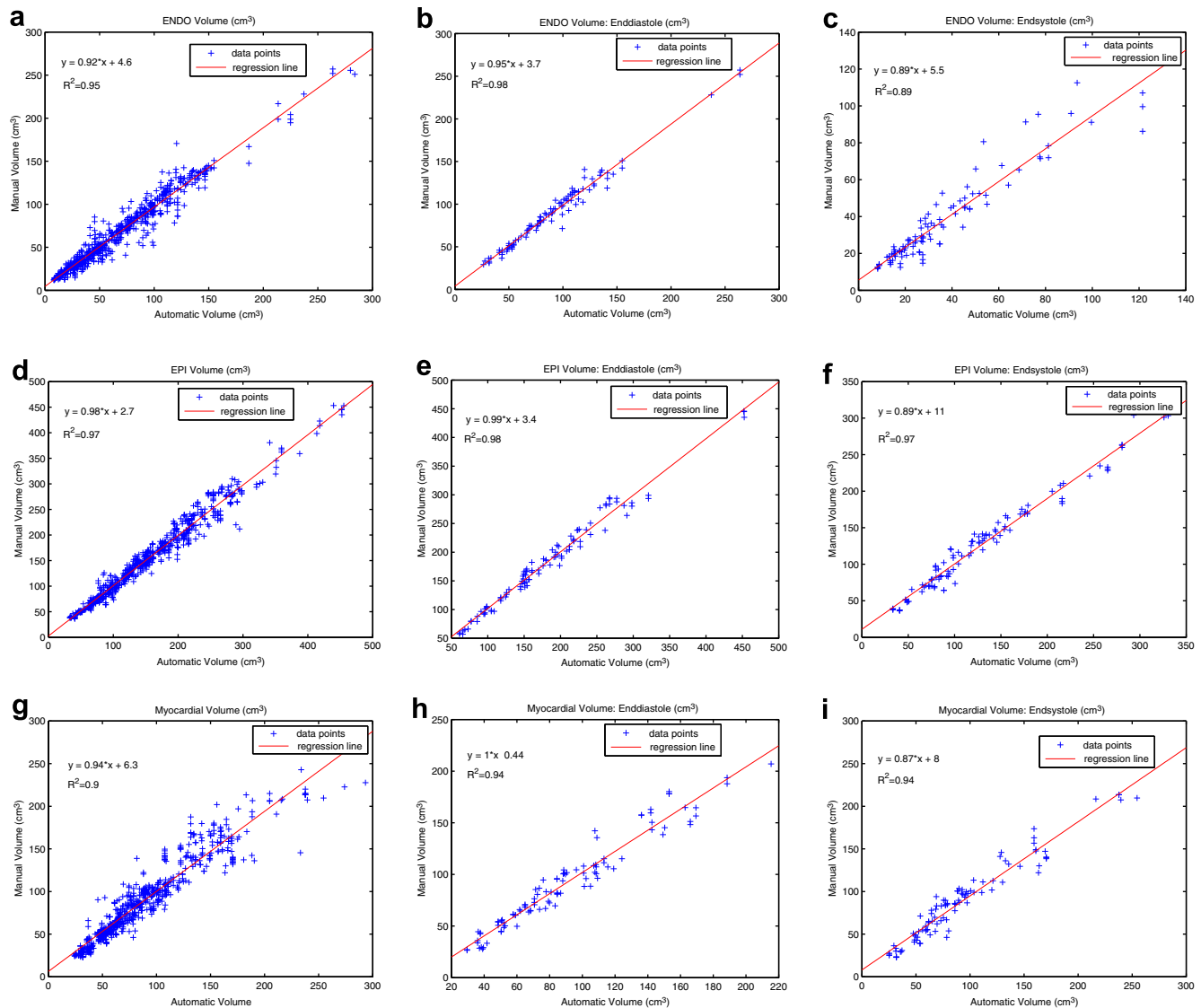


Fig. 9. Regression graph of the IC9-based results after using the 2D + time algorithm (horizontal axis) versus the ground truth volume (vertical axis). End-diastolic/end-systolic volume regression graphs were calculated using the three frames closest to end-diastole/end-systole. (a–c): volume inside the endocardium; (d–f): volume inside the epicardium; (g–i): myocardial volume. The compared volumes of each datapoint use the volume regions corresponding to the image slices that were both manually segmented and automatically segmented using our algorithm.

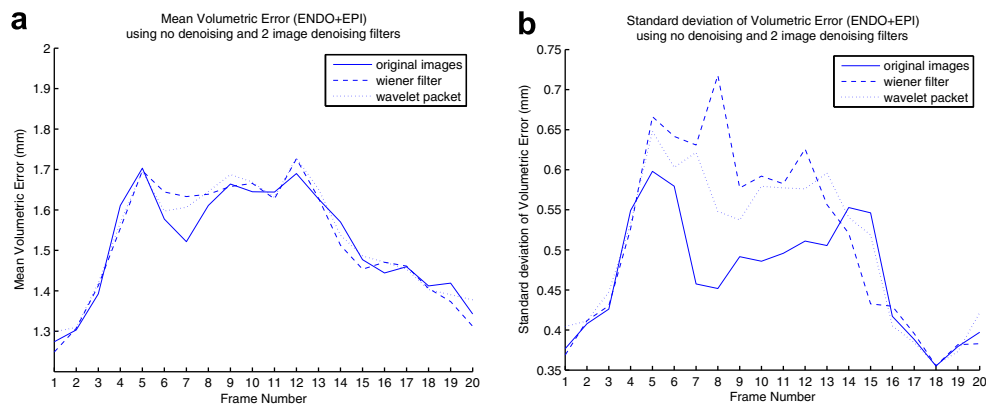


Fig. 10. Results after using IC9 with various denoising algorithms. (a) The mean volumetric error. (b) The volumetric error's standard deviation. There is no noticeable improvement in the error measures, most likely due to the modest blurring of the myocardial borders caused by the denoising algorithms.

Table 3
Results of other 3D cardiac segmentation algorithms

	Endocardium	Epicardium
Mitchell et al. (2002)	2.75 ± 0.86	2.63 ± 0.76
Kaus et al. (2004)	2.28 ± 0.93	2.62 ± 0.75
Lorenzo-Valdés et al. (2004)	1.88 ± 2.00	2.75 ± 2.62
Lötjönen et al. (2005)	2.01 ± 0.31	2.77 ± 0.49
Tölli et al. (2006)	1.77 ± 0.36	1.85 ± 0.51

Note that most algorithms are tested on different datasets using slightly different error metrics from each other. Also many of the algorithms have been tested on different frames of the cardiac cycle. Please refer to the corresponding papers for more details.

the error of IC9 and IC3 increases significantly. However, in frame 6, which is the first frame where we use the model trained on systole data, the error drops rapidly. This indicates that inverse compositional-based algorithms tend to become ill-conditioned when trying to segment frames using models built on deviating frames. To test this hypothesis we segmented frames 4, 5 using the same data that was used to segment frames 6–15 and this error jump disappeared as expected. From Fig. 7, we see that IC9 performs significantly better than IC3 does near the end-systole frames. This demonstrates that a cascade of models applied to each frame can improve significantly our segmentation results. The results are sufficiently encouraging to warrant more research on similar approaches.

By comparing the Gauss–Newton optimization to the Classical approach we notice that though Gauss–Newton performs in general better, the difference is not as pronounced as one might have expected. We take this as an indication that as one begins to delve in the range of at least 50 optimization parameters, as is common with 3D and 4D AAMs, standard optimization algorithms easily fall into local minima, making the optimization problem a major bottleneck in fitting such high dimensional models.

The previously described intensity normalizing mechanism requires a manual segmentation of a number of images at the end-diastolic frame of the image sequence

in order to build an intensity correcting linear transformation. This manual segmentation is also used to initialize the 3D AAM. One can argue that the above mentioned approach is problematic, since the initialization of the model can be time consuming. Often the borders are not very clear – making the manual segmentation slow and error prone – and the intensities are not always homogeneous. We view a number of ways of dealing with poor initializations. Since the inverse compositional-based algorithm is quite fast, given any initialization of our model, we could perform multiple optimizations of the 3D model using a number of neighboring locations as starting positions for the model. The correct initialization would be the one that leads to the best fit of the model’s appearance on the images. We could also investigate greedily modifying the length of the updating vectors provided by the AAM optimization algorithms, which is something we have not attempted to do in this paper. By multiplying the updating vectors by a number of constants, the fitting algorithms will not fall into local minima as easily. Such approaches have been previously used to deal with the model initialization problem (Mitchell et al., 2002). To deal with the intensity normalization, the user would simply have to select a number of endocardial and myocardial regions – not necessarily close to the myocardial borders – that could be sampled to determine the linear transformation for the intensity normalization. This would be quite fast, since the user would not necessarily have to position the contours close to the myocardial borders. He would just need to indicate whether each selected region is inside the myocardial region, or inside the endocardial region.

We also experimented with two popular denoising algorithms before performing any training or fitting with the 3D AAM. The results are shown in Fig. 10. We denoised the images with an adaptive 2D Wiener filter, using a 3×3 neighborhood to estimate each neighborhood’s mean and standard deviation. We also denoised the image with a 2D wavelet packet-based approach (Mallat, 1999). We noticed virtually no difference in the segmentation accuracy and in a few cases we even observed a small degradation in

Table 4

A comparison of the ENDO + EPI volumetric errors of our hierarchical wavelet-based 2D + time ASM vs. a standard PCA-based 2D + time ASM (wavelet-based/standard PCA)

Fitting algorithm	25% of data	50% of data	75% of data	100% of data
IC9	$1.496 \pm 0.467/1.505 \pm 0.489$ *** (0)	$1.485 \pm 0.459/1.497 \pm 0.482$ *** (0)	$1.483 \pm 0.452/1.488 \pm 0.470$ ×(0.118)	$1.473 \pm 0.445/1.480 \pm 0.463$ *(0.022)
IC3	$1.586 \pm 0.568/1.602 \pm 0.590$ *** (0)	$1.569 \pm 0.557/1.583 \pm 0.575$ *** (0)	$1.562 \pm 0.542/1.568 \pm 0.562$ *(0.026)	$1.552 \pm 0.535/1.564 \pm 0.557$ *** (0)
Gauss–Newton	$1.590 \pm 0.765/1.608 \pm 0.785$ *** (0)	$1.573 \pm 0.742/1.593 \pm 0.764$ *** (0)	$1.560 \pm 0.727/1.579 \pm 0.751$ *** (0)	$1.543 \pm 0.706/1.563 \pm 0.738$ *** (0)
Classical	$1.685 \pm 0.587/1.710 \pm 0.610$ *** (0)	$1.667 \pm 0.581/1.688 \pm 0.598$ *** (0)	$1.655 \pm 0.563/1.667 \pm 0.579$ *** (0)	$1.642 \pm 0.558/1.658 \pm 0.576$ *** (0)

The hierarchical approach is consistently better across all frames and across all algorithms used to fit the 3D AAM. We attribute this statistically significant difference to the hierarchical model’s improved ability to model local deformations in the presence of small data sets. As indicated in the text, our full data set consisted of 66 training samples, and we demonstrate the results for various random subsets of this data set (25%, 50%, 75% and 100% of the original training set). * $0.01 \leq p < 0.05$, ** $0.001 \leq p < 0.01$, *** $0 \leq p < 0.001$, × $0.05 \leq p$.

the segmentation accuracy-attributed to a small blurring in the myocardial borders. This validated previous research regarding the robustness of simple AAMs in dealing with noisy images. When using AAMs, denoising algorithms are necessary only with extremely noisy images – which was not the case with our dataset. Similarly, we noticed virtually no improvement when using the denoised images to improve the 2D + time phase of the segmentation.

By inspecting Table 2 and Fig. 8, we make the following observations. In all cases the 2D + time approach offers an improvement in terms of mean error and standard deviation. In most cases the improvement is also statistically significant. The 2D + time approach never leads to worse results, indicating that it offers a reliable approach for improving the 3D AAM-based segmentation, while simultaneously incorporating temporal knowledge in our segmentation. As expected, by observing Fig. 8, the better the 3D AAM-based segmentation is, the smaller the improvement provided by the 2D + time approach.

Table 4 demonstrates that the wavelet-based hierarchical approach provides a statistically significant – though modest – improvement in the segmentation compared to standard PCA-based methods. This modest improvement is due to the greater local shape variability offered by the wavelet-based approach, especially for small training set where standard PCA-based approaches provide very few modes of deformation. A general observation about the 2D + time approach is that in no case did we observe a degradation in our results. Finally, a qualitative observation based on a visual inspection of our results demonstrates that after applying the 2D + time approach the contour motion across time is much more uniform and natural. We also experimented with the method described in van Ginneken et al. (2002). This landmark placement method provided results that were very similar to those obtained using the procedure described in Section 2.2.3. On close inspection of the features selected with the van Ginneken et al. (2002) method after its feature selection step, we noticed that the strongest features selected were based on the first derivative and effectively searched for the strongest edge like other methods do (see Section 2.2.3 and Cootes and Taylor, 2004). This explains why we obtained similar results with both methods. The volume regression graphs in Fig. 9 demonstrate improved volume estimation compared to previously published work (Mitchell et al., 2002), with the endocardial volume at end-systole being the most difficult to estimate.

6. Conclusions

We have proposed a reliable model for the analysis of short axis cardiac MRI. We first presented an algorithm for fitting 3D active appearance models on short axis cardiac MR images, and observed an almost 44-fold improvement in the segmentation speed and a segmentation accuracy that is on par (and often better) with Gauss–Newton optimization, the most widely used algorithm for such

optimization problems. We then showed how a 2D + time hierarchical shape model could significantly improve our results. The model integrates temporal knowledge in the segmentation process and makes localized adjustments to the segmentation that improve the 3D AAM segmentation. The results lead us to believe that an approach similar to the one we have proposed here, can significantly improve the reliability of cardiac image analysis algorithms. A greedy approach for adjusting the length of the update parameters Δp , Δq during the 3D AAM optimization might lead to better results, and it is worth investigating as future research. We presented encouraging results demonstrating that a cascade of cardiac models can improve our segmentation results. Algorithms similar to the ones we presented in this paper make it feasible to fit hundreds of models within reasonable time limits and with a minimal loss in accuracy and reliability. The computer-based analysis of medical images can complement the manual and labor intensive approaches used today, and will most likely improve medical diagnosis leading to faster and more accurate detection and treatment of cardiovascular disease.

Acknowledgements

We thank Dr. Paul Babyn and Dr. Shi-Joon Yoo of the Department of Diagnostic Imaging at the Hospital for Sick Children in Toronto for providing us the MRI data. J.K.T. holds the Canada Research Chair in Computational Vision and gratefully acknowledges its financial support. A.A. would also like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for its financial support through the PGS-M/PGS-D scholarship program. We also thank the reviewers for their insightful comments and suggestions.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at [doi:10.1016/j.media.2007.12.003](https://doi.org/10.1016/j.media.2007.12.003).

References

- American Heart Association, 2004. International Cardiovascular Disease Statistics. <<http://www.americanheart.org>> (Online).
- Amini, A.A., Duncan, J.S., 1992. Bending and stretching models for LV wall motion analysis from curves and surfaces. *Image Vis. Comput.* 10 (6), 418–430.
- Baker, S., Matthews, I., 2001. Equivalence and efficiency of image alignment algorithms. In: *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 1090–1097.
- Baker, S., Goss, R., Matthews, I., 2004. Lucas-Kanade 20 years on: a unifying framework. *Int. J. Comput. Vis.* 56 (3), 221–255.
- Bardinet, E., Ayache, N., Cohen, L.D., 1996. Tracking and motion analysis of the left ventricle with deformable superquadrics. *Med. Image Anal.* 1 (2), 129–150.
- Benayoun, S., Ayache, N., 1998. Dense and nonrigid motion estimation in sequences of medical images using differential constraints. *Int. J. Comput. Vis.* 26 (1), 25–40.

- Bosch, J.G., Mitchell, S.C., Lelieveldt, B.P.F., Nijland, F., Kamp, O., Sonka, M., 2002. Automatic segmentation of echocardiographic sequences by active appearance motion models. *IEEE Trans. Med. Imaging* 21 (11), 1374–1383.
- Cohen, L.D., Cohen, I., 1992. Deformable models for 3D medical images using finite elements and balloons. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June, pp. 592–598.
- Cohen, I., Cohen, L., Ayache, N., 1991. Introducing new deformable surfaces to segment 3D images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June, pp. 738–739.
- Cootes, T., Taylor, C., 1995. Combining point distribution models with shape models based on finite element analysis. *Image Vis. Comput.* (13), 403–409.
- Cootes, T.F., Taylor, C., 1998. Active appearance models. In: *Proceedings of the European Conference on Computer Vision*, vol. 2, pp. 484–498.
- Cootes, T.F., Taylor, C.J., 2004. Statistical models of appearance for computer vision. Tech. rep., *Imaging Science and Biomedical Engineering*, University of Manchester.
- Davatzikos, C., Tao, X., Shen, D., 2003. Hierarchical active shape models, using the wavelet transform. *IEEE Trans. Med. Imaging* 22 (3), 414–422.
- de Bruijne, M., van Ginneken, B., Viergever, M., Niessen, W., 2003. Adapting active shape models for 3D segmentation of tubular structures in medical images. In: Taylor, C., Noble, A. (Eds.), *Information Processing in Medical Imaging*, pp. 136–147.
- Delhay, B., Lotjonen, J., Clarysse, P., Katila, T., Magnin, I., 2005. A dynamic 3D cardiac surface model from MR images. *Comput. Cardiol.*
- Frangi, A.F., Niessen, W., Viergever, M.A., 2001. Three-dimensional modeling for functional analysis of cardiac images: a review. *IEEE Trans. Med. Imaging* 20 (1), 2–25.
- Gering, D.T., 2003. Automatic segmentation of cardiac MRI. In: *Lecture Notes in Computer Science*, vol. 2878, October, pp. 524–532.
- Hamarneh, G., Gustavsson, T., 2004. Deformable spatio-temporal shape models: extending ASM to 2D + time. *Image Vis. Comput.* 22 (6), 461–470.
- Kaus, M.R., von Berg, J., Weese, J., Niessen, W., Pekar, V., 2004. Automated segmentation of the left ventricle in cardiac MRI. *Med. Image Anal.* 8 (3), 245–254.
- Lorenzo-Valdés, M., Sanchez-Ortiz, G.I., Elkington, A.G., Mohiaddin, R.H., Rueckert, D., 2004. Segmentation of 4D cardiac MR images using a probabilistic atlas and the EM algorithm. *Med. Image Anal.* 8 (3), 255–265.
- Lötjönen, J., Kivistö, S., Koikkalainen, J., Smutek, D., Lauerma, K., 2004. Statistical shape model of atria, ventricles and epicardium from short- and long-axis MR images. *Med. Image Anal.* 8 (3), 371–386.
- Lötjönen, J., Antila, K., Lamminmaki, E., Koikkalainen, J., Lilja, M., Cootes, T., 2005. Artificial enlargement of a training set for statistical shape models: application to cardiac images. In: *Proceedings of the FIMH 2005*.
- Maintz, J., Viergever, M., 1998. A survey of medical image registration. *Med. Image Anal.* 2 (1), 1–36.
- Mallat, S., 1999. *A Wavelet Tour of Signal Processing*, fifth ed. Academic Press.
- Matthews, I., Baker, S., 2004. Active appearance models revisited. *Int. J. Comput. Vis.* 60 (2), 135–164.
- McInerney, T., Terzopoulos, D., 1995. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to 4D image analysis. *Comput. Med. Imaging Graph.* 19 (1), 69–83.
- Mitchell, S.C., Lelieveldt, B.P.F., van der Geest, R., Bosch, H.G., Reiber, J.H.C., Sonka, M., 2001. Multistage hybrid active appearance model matching: segmentation of left and right ventricles in cardiac MR images. *IEEE Trans. Med. Imaging* 20 (5), 415–423.
- Mitchell, S.C., Bosch, J.G., Lelieveldt, B.P.F., van der Geest, R.J., Reiber, J.H.C., Sonka, M., 2002. 3D active appearance models: segmentation of cardiac MR and ultrasound images. *IEEE Trans. Med. Imaging* 21 (9), 1167–1178.
- Nain, D., Haker, S., Bobick, A., Tannenbaum, A., 2006. Shape-driven 3D segmentation using spherical wavelets. In: *Proceedings of the MICCAI 2006*.
- Panfilov, A.V., Holden, A.V. (Eds.), 1996. *Computational Biology of the Heart*. John Wiley and Sons.
- Papademetris, X., 2000. Estimation of 3D left ventricular deformation from medical images using biomechanical models. Ph.D. Thesis, Yale University, Department of Diagnostic Radiology.
- Papademetris, X., Sinusas, A., Dione, D., Constable, R., Duncan, J., 2002. Estimation of 3D left ventricular deformation from medical images using biomechanical models. *IEEE Trans. Med. Imaging* 21 (7), 786–800.
- Park, J., Metaxas, D.N., Young, A.A., Axel, L., 1996. Deformable models with parameter functions for cardiac motion analysis from tagged MRI. *IEEE Trans. Med. Imaging* 15, 278–289.
- Pentland, A., Horowitz, B., 1991. Recovery of nonrigid motion and structure. *IEEE Trans. Pattern Anal.* 13 (July), 730–742.
- Rosse, C., Gaddum-Rosse, P., 1997. *Hollinshead's Textbook of Anatomy*, fifth ed. Lippincott-Raven.
- Rousson, M., Paragios, N., Deriche, R., 2004. Implicit active shape models for 3D segmentation in MRI imaging. In: *Proceedings of the MICCAI 2004*.
- Sermesant, M., Forest, C., Pennec, X., Delingette, H., Ayache, N., 2003. Deformable biomechanical models: application to 4D cardiac image analysis. *Med. Image Anal.* 7 (4), 475–488.
- Shen, D., Davatzikos, C., 2000. An adaptive-focus deformable model using statistical and geometric information. *IEEE Trans. Pattern Anal.* 22 (8), 906–913.
- Shen, D., Davatzikos, C., 2001. An adaptive-focus statistical shape model for segmentation and shape modeling of 3D brain structures. *IEEE Trans. Med. Imaging* 20 (4), 257–270.
- Stegmann, M.B., 2004. Generative interpretation of medical images. Ph.D. Thesis, Technical University of Denmark.
- Stegmann, M.B., Ólafsdóttir, H., Larsson, H.B.W., 2005. Unsupervised motion-compensation of multi-slice cardiac perfusion MRI. *Med. Image Anal.* 9 (4), 394–410.
- Tölli, T., Koikkalainen, J., Lauerma, K., Lötjönen, J., 2006. Artificially enlarged training set in image segmentation. In: *Proceedings of the MICCAI 2006*.
- van Assen, H.C., Danilouchkine, M.H., Frangi, A.F., Ordás, S., Westenberg, J.J., Reiber, J.H., Lelieveldt, B.P., 2006. SPASM: a 3D-ASM for segmentation of sparse and arbitrarily oriented cardiac MRI data. *Med. Image Anal.* 10, 286–303.
- van Ginneken, B., Frangi, A.F., Staal, J., ter Haar Romeny, B.M., Viergever, M.A., 2002. Active shape model segmentation with optimal features. *IEEE Trans. Med. Imaging* 21 (8), 924–933.
- Yan, P., Lin, N., Sinusas, A.J., Duncan, J.S., 2005. A boundary element-based approach to analysis of LV deformation. In: *Proceedings of the MICCAI 2005*.
- Zitova, B., Flusser, J., 2003. Image registration methods: a survey. *Image Vis. Comput.* 21, 977–1000.