Medical Image Analysis 14 (2010) 70–84

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Optimizing boundary detection via Simulated Search with applications to multi-modal heart segmentation

J. Peters *, O. Ecabert, C. Meyer, R. Kneser, J. Weese

Philips Research Europe – Aachen, 52066 Aachen, Germany

ARTICLE INFO

Article history:
Received 12 November 2008
Received in revised form 13 October 2009
Accepted 15 October 2009
Available online 22 October 2009

Keywords:
Multi-modal image segmentation
Cardiac segmentation
Shape-constrained deformable models
Boundary detection
Simulated Search

ABSTRACT

Segmentation of medical images can be achieved with the help of model-based algorithms. Reliable boundary detection is a crucial component to obtain robust and accurate segmentation results and to enable full automation. This is especially important if the anatomy being segmented is too variable to initialize a mean shape model such that all surface regions are close to the desired contours. Several boundary detection algorithms are widely used in the literature. Most use some trained image appearance model to characterize and detect the desired boundaries. Although parameters of the boundary detection can vary over the model surface and are trained on images, their performance (i.e., accuracy and reliability of boundary detection) can only be assessed as an integral part of the entire segmentation algorithm. In particular, assessment of boundary detection cannot be done locally and independently on model parameterization and internal energies controlling geometric model properties.

In this paper, we propose a new method for the local assessment of boundary detection called Simulated Search. This method takes any boundary detection function and evaluates its performance for a single model landmark in terms of an estimated geometric boundary detection error. In consequence, boundary detection can be optimized per landmark during model training. We demonstrate the success of the method for cardiac image segmentation. In particular we show that the Simulated Search improves the capture range and the accuracy of the boundary detection compared to a traditional training scheme. We also illustrate how the Simulated Search can be used to identify suitable classes of features when addressing a new segmentation task. Finally, we show that the Simulated Search enables multi-modal heart segmentation using a single algorithmic framework. On computed tomography and magnetic resonance images, average segmentation errors (surface-to-surface distances) for the four chambers and the trunks of the large vessels are in the order of 0.8 mm. For 3D rotational X-ray angiography images of the left atrium and pulmonary veins, the average error is 1.3 mm. In all modalities, the locally optimized boundary detection enables fully automatic segmentation.

1. Introduction

Automatic segmentation of anatomical structures in medical images is a key task for data analysis and visualization. Deformable or parametric shape models are widely used to address this task (see, e.g., Kass et al. (1988), Staib and Duncan (1992), Cootes et al. (1994), McInerney and Terzopoulos (1996), Cootes and Taylor (2001), Weese et al. (2001), Mitchell et al. (2001), van Ginneken et al. (2002), Kaus et al. (2003), de Bruijne et al. (2003), Heimann et al. (2007), van Assen et al. (2008)). These approaches deform a surface model by minimizing an energy function: an external energy drives the surface model to the visible organ boundaries in the image. For parametric shape models, the space of allowed shapes is predetermined. For deformable models, an internal energy keeps the surface smooth or close to expected shapes. Two approaches can be distinguished (see, e.g., McInerney and Terzopoulos (1996), Sonka and Fitzpatrick (2000)): boundary-driven segmentation detects visible boundaries and attracts the surface to the detected locations. Region-based approaches aim at a segmentation that best explains the complete image or large regions thereof. This paper follows the approach of boundary-driven segmentation.

Robust and accurate boundary detection is a crucial point to achieve fully automatic and accurate segmentation results. A key problem is the abundance of misleading image structures that need to be discriminated from the correct boundaries of the organs being segmented. To be robust for very approximate model

---

* Corresponding author. Tel.: +49 241 6003 580; fax: +49 241 6003 442.
E-mail addresses: jochen.peters@philips.com (J. Peters), olivier.ecabert@philips.com (O. Ecabert), carsten.meyer@philips.com (C. Meyer), reinhard.kneser@philips.com (R. Kneser), juergen.weese@philips.com (J. Weese).

1361-8415/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.media.2009.10.004

initialization, correct boundaries must be detected within a large capture range. During model convergence, the focus shifts to accurate boundary matching.

A general scheme for boundary detection is as follows: for a set of surface points (landmarks) the image neighborhood is searched for target points maximizing some match function. Typically, per landmark, target points are searched along a 1-dimensional sampling profile perpendicular to the model surface. Crucial for robust and accurate boundary detection is the design (functional form) and parameterization of the match function. Ideally, the match function should attain its maximum within the tested search range at the correct boundary. If the anatomical boundaries are not visible no other target points should be found and all sampling points should be rejected.

Different approaches to design the match function and to estimate the parameters have been published. All functions are based on some local image features like gradients, gray values, or texture-related quantities. Per sampling point within the search range, one or several of these features are evaluated and one scalar match value is calculated from the feature values. The sampling point with maximum match is chosen as target point or may be rejected if the match falls below some threshold.

Most approaches to train the match functions for a given application first define the functional form and then determine the free parameters based on training images and given reference segmentations. Depending on the complexity of the task, parameters may be determined globally or with spatial variations to account for varying boundary appearance over the model surface. To assess the overall quality of the resulting match functions, new images can be segmented and segmentation errors can be measured. This assessment, however, suffers from two weaknesses: first, it can only be applied after having trained and assigned the match functions for all landmarks. Intermediate evaluation per landmark is not possible. Second, since the segmentation process always includes contributions from the internal energy to regularize the surface shape, local errors are not strictly correlated with the boundary detection errors. Missed boundaries may be hidden by the shape interpolation, and correct detections may be overruled by neighboring misdetections in combination with shape constraints.

To overcome these shortcomings, we propose to use the Simulated Search method which defines a dedicated local performance measure. This measure evaluates the expected boundary detection error in a continuous, geometric sense for any given match function and any single landmark. It thus decouples the performance of the boundary detection from the final segmentation algorithm and eliminates the obscuring influence of the shape model. The idea of the Simulated Search was first published in Peters et al. (2005) and has subsequently been adopted by Heimann et al. (2007), Heimann et al. (2007) as local performance measure for already trained models. Based on this local performance measure we propose to locally optimize the match functions. More specifically, we offer a (large) variety of match functions without prior assignment to the landmarks. Based on the estimated boundary detection errors, we then select the best match function per triangle of a surface mesh model.

To demonstrate the success of the proposed method, we will address the exemplary task of cardiac segmentation in multiple imaging modalities. The ability to locally optimize the match functions and to extend the capture range is of particular importance for two reasons: first, due to shape variations between patients and across cardiac phases, a complex mean mesh comprising multiple cardiac structures cannot, in general, be initialized close to all wanted contours in a new image. Hence, initialization (manual or automatic) is typically only approximate and boundaries need to be captured over a large search range. Second, the image appearance in different cardiac regions varies. Depending on the imaging protocol, blood pool contrast may exhibit small or large variations. Similar observations apply to thickness and appearance of tissue surrounding the blood pool. Allowing for variable match function design (e.g., including or excluding features extracted from the blood pool) and parameterization across the model surface is crucial for proper boundary detection and the discrimination against false edges.

The major contributions of this paper are: (1) the Simulated Search is presented with an in-depth quantitative analysis of the obtained improvements of capture ranges and final accuracies. (2) Examples illustrate the selection behaviour of the Simulated Search and show how the method can guide the user in the design of new match functions. (3) For the exemplary task of cardiac segmentation, we demonstrate successful multi-modal image segmentation within one algorithmic framework. Here, the Simulated Search is the key to adjust the boundary detectors to new modalities. Segmentation results will be discussed in the context of other work on cardiac segmentation using model-based adaptation and atlas-based registration approaches by Frangi et al. (2001), Mitchell et al. (2001), Kaus et al. (2004), Lötjönen et al. (2004), van Assen et al. (2006), Lorenz and von Berg (2006), Zheng et al. (2008), Wierzbicki et al. (2008), Zhuang et al. (2008), van Assen et al. (2008). An overview of that work will be presented together with our results.

The paper is organized as follows: Section 2 summarizes the state-of-the-art for match function design (Mahalanobis distance, edge detectors enhanced by additional features, various classifiers) and parameter estimation. Section 3 introduces our local performance measure based on the Simulated Search. Section 4 introduces image data, model structures, and error metrics used in our experiments which are covered in Section 5. Section 6 concludes the paper.

2. Match functions: state-of-the-art

2.1. Mahalanobis distance

Image appearance at the wanted boundaries can be described within a multi-dimensional feature space. Typically, a 1-dimensional profile of gray values or gradients across the model surface forms a so-called feature vector $\boldsymbol{f}$. Assuming a uni-modal, multivariate normal distribution of these vectors in the corresponding vector space, the maximum likelihood position within the search space corresponds mathematically to the minimization of the negative log-likelihood which is proportional to the Mahalanobis distance

$$d_{\text{Mahalanobis}}(\boldsymbol{f}) = (\boldsymbol{f} - \boldsymbol{f}_{\text{mean}})^{\top} \cdot \Sigma^{-1} \cdot (\boldsymbol{f} - \boldsymbol{f}_{\text{mean}}) \qquad (1)$$

where $\boldsymbol{f}_{\text{mean}}$ and Σ are the mean vector and the covariance matrix of the distribution.

In Cootes et al. (1994), Cootes and Taylor (2001), Mitchell et al. (2001), Heimann et al. (2006), Heimann et al. (2007), Brejl and Sonka (2000), Tobon-Gomez et al. (2008), feature vectors are composed of plain gray values and/or differences of gray values along the profile to compensate for arbitrary offsets. Furthermore, additional calibration to compensate for varying gray value ranges is often achieved by normalizing the resulting feature vectors prior to the statistical analysis. Normalization can simply scale the vector to unit length, or it can affect individual features that are calibrated using global image statistics.

Defining the parameters for (1) is straightforward: $\boldsymbol{f}_{\text{mean}}$ and Σ are calculated from the feature vectors $\boldsymbol{f}$ of the training examples.

2.2. Edge detectors with additional constraints

If visible boundaries are characterized by (strong) gradients then edge detection is a natural starting point. In Kass et al. (1988), Staib and Duncan (1992), Cohen and Cohen (1993), Cootes et al. (1995), Chakraborty et al. (1996), Sakalli et al. (2006), the gradient magnitude alone is used as match function. In Weese et al. (2001), Kaus et al. (2003), the orientation of the gradient with respect to the outward surface normal is taken into account to differentiate between bright-to-dark and inverse transitions. To better characterize the appearance of wanted boundaries, constraints on additional features like local gray values and gradient sizes are added in Delingette (1999), Montagnat et al. (2003), Sermesant et al. (2003), Kaus et al. (2004), Sermesant et al. (2006). If any constraint's feature value violates some learned acceptance interval, the edge is rejected by setting the match value to 0.

Here, free parameters may include the expected orientation of the gradient (with respect to the model surface) and the constraints' acceptance intervals. The orientation can be given a priori or learned from training examples. Acceptance intervals for each constraining feature can be derived from training examples using some heuristic rules (e.g., based on mean and standard deviation of the feature values observed at the reference boundaries).

2.3. kNN classifiers

Several authors point out that the above approach considers only correct learning examples since the distribution is trained only from feature vectors evaluated at reference positions. Furthermore, the assumption of a uni-modal, multi-variate normal distribution may be false. To train a more general classifier that is able to distinguish between correct and false positions, a k-nearest-neighbor (kNN) classifier is trained as follows. Feature vectors $\boldsymbol{f}$ are evaluated and stored for two classes containing (1) reference positions and (2) false positions which are typically generated by shifting the references along the surface normal. To classify a new feature vector $\boldsymbol{f}$ during segmentation, the likelihood to belong to either class is then estimated from the relative classification counts of the k-nearest-neighbors of $\boldsymbol{f}$ among all stored vectors.

In de Bruijne et al. (2003), Heimann et al. (2007), Heimann et al. (2007), feature vectors are again composed of gray values or differences thereof, possibly again including some normalization steps.

The idea of kNN classification has also been applied in a hybrid combination of region-based and boundary-driven segmentation in (van Ginneken et al., 2002; de Bruijne and Nielsen, 2004). First, a classifier is trained not for boundary versus non-boundary positions but for voxels from two tissue classes (inside versus outside region). Boundary detection is then performed by shifting the model surface within the search range until the joint classification probability of the voxels inside and outside is maximized. Thereafter, the detected boundary is used to attract the model surface.

For kNN classifiers, all statistics from the training examples are contained in the stored feature vectors and their classification. Here, k is a free parameter of the classifier. Typically, k is set globally to some fixed value.

2.4. Other classifiers

Recently, a variety of publications with different approaches for boundary detection has appeared that do not completely fit into the above schemes. van Assen et al. (2006, 2008) applies a fuzzy voxel classification into three classes (bright, medium, dark) based on relative gray values. Adjustable thresholds enable a classification for different imaging modalities. Combining fuzzy classifications of multiple voxels and applying a majority voting scheme, positions along a search profile are classified into three tissues. Finally, myocardial boundaries are extracted by predefined rules from tissue transitions.

Zheng et al. (2008), Zhang et al. (2008), Sukno et al. (2007), Sukno and Frangi (2008) resume the idea of discriminative training to distinguish between correct and false positions. The approaches start with a set of local image features derived from gray values, gradients, and second order derivatives, or Haar wavelets, partly converted into rotation-invariant quantities. During segmentation, these features are sampled in the neighborhood of the tested boundary positions. In Zheng et al. (2008), Zhang et al. (2008), weak classifiers for correct versus false boundary examples are trained for each feature type. These are then combined via probabilistic boosting trees into a strong classifier. In Sukno et al. (2007), Sukno and Frangi (2008), the work from van Ginneken et al. (2002) is resumed and several changes are introduced: multiscale Gaussian derivatives are replaced by rotation-invariant features. kNN classifiers are replaced by multivalued neurons. These accelerate the calculation as compared to kNN classifiers. Furthermore, they are used to classify input features from a given image location into classes describing increasing distances from the reference boundary. This replaces the inside–outside voxel classification of van Ginneken et al. (2002). The system is thus trained to locally estimate the distance to the correct boundary along the search profile. Boundary detection is performed by looking for a "V" shaped minimum of the estimated boundary distances.

2.5. Spatially varying parameters and clustering

Depending on the application, appearance may vary over the model surface. Shape models with correspondence preserving landmarks allow to train local match functions. E.g., $\boldsymbol{f}_{\text{mean}}$ and Σ, feature vectors for kNN classifiers, or acceptance intervals can be evaluated and stored per landmark. The resulting models, however, typically suffer from insufficient statistics of sparse training examples since the necessary effort involved in ground truth generation limits the training data bases.

To exploit a larger statistical training basis – but nevertheless allow for locally varying parameterization – landmarks can be clustered into regions of similar appearance according to the used feature vectors, see, e.g., Heimann et al. (2007), Heimann et al. (2007), Brejl and Sonka (2000), Kaus et al. (2004). In van Assen et al. (2006), model regions are defined to reflect different appearances due to coil inhomogeneities. In all approaches, parameters are then estimated per region and all landmarks per region share the same parameterization. For the Mahalanobis distance approach, clusters may represent a mixture of Gaussians, thus relaxing the assumption of one single multi-variate normal distribution.

2.6. Local optimization

In van Ginneken et al. (2002), an explicit local optimization scheme is presented. The approach follows the hybrid region-based and boundary-driven segmentation scheme using kNN voxel classifiers (see Section 2.3). This approach departs from using fixed feature vectors and it does not cluster landmarks to estimate spatially varying parameters. Rather, multiscale Gaussian derivatives of the local gray value neighborhood (also known as local jet) are used to derive a variety of potentially useful features. From all proposed features, a subset is then learned by feature selection techniques such that a kNN classifier using the selected features performs optimally on the classified training voxels. This is done per landmark. Similarly, Sukno et al. (2007), Sukno and Frangi (2008) train their distance classifiers per model landmark.

Local optimization via the Simulated Search directly evaluates the boundary detection and estimates geometric errors which are then minimized. In contrast, van Ginneken et al. (2002), Sukno et al. (2007), Sukno and Frangi (2008) optimize voxel or distance classifiers and boundary detection is only indirectly performed therefrom.

3. Performance measure via Simulated Search

3.1. Motivation

As stated in the introduction, evaluating a trained model – with match functions assigned to the model landmarks – in terms of the resulting segmentation errors suffers from two weaknesses: evaluating the quality of the boundary detection per landmark is not possible since all boundary detectors are applied simultaneously and the internal energy obscures local detection errors. Furthermore, exploring possible improvements if an assigned match function for some landmark is testwise replaced by some alternative function is computationally prohibitive if a large set of alternative functions shall be tested for all landmarks. This would involve a new segmentation for each replacement for each landmark.

An assessment of the boundary detection alone per landmark and for any proposed match function enables a local optimization that is not possible in the above evaluation framework. The performance measure defined below by the Simulated Search estimates the expected boundary detection error in a continuous, geometric sense for any proposed function per landmark. Based on this measure, a large list of candidate functions can be evaluated and the best function can be selected per landmark. The candidate list may include match functions using different functional forms, different features, and a variety of parameter settings.

For complex organs such as the heart with multiple structures with varying appearance, using different image features in different regions may indeed improve the boundary characterization (detection and discrimination against competing boundaries). E.g., gray values in the blood pool may vary. Depending on the contrast protocol and imaging parameters, variations in one region may be larger than in another region (e.g., some clinical applications focus on left chamber contrast and right chamber contrast can be highly variable). Features based on these gray values may be useful in some regions but harmful elsewhere. Similarly, gray values of outside tissue (myocardium, fat, lung, etc.) may be predictable if tissue thickness and appearance are similar across patients. In some regions, including such features may again be harmful. Apart from selecting the locally most suitable features, the best parameterization may change from one region to another. Deciding – for all landmarks – on the best combination of features and parameters is hardly feasible without some performance measure. The Simulated Search method provides the basis for this decision.

3.2. Simulated Search

The central idea is to simulate the boundary detection within the training images and to record the errors with respect to given reference positions. The simulation disturbs the pose (position and orientation) of each landmark around the given reference pose to reflect expected poses during adaptation. For each perturbed pose, a target point is searched (hence the method's name). This is done per landmark which decouples the boundary detection from regularizing internal energy terms and thus from neighboring landmarks. Errors are measured as distances between detected target points and reference boundaries. This measurement reflects the ultimate goal to minimize the distances between segmented and reference surfaces. Furthermore, recording continuous distances instead of binary (boundary versus non-boundary) classification errors relaxes the assumption made by kNN classifiers that boundary positions can be annotated with infinite accuracy. Here, target points close to the provided reference segmentation are rated as good (the closer the better) while distant target points are rated as bad. In practice, reference segmentations may indeed deviate slightly from the true boundary and rating minor offsets as "still good" relaxes quality requirements on the reference segmentation.

For the concrete formalism, let $\boldsymbol{x}_i$ and $\boldsymbol{n}_i$ denote the coordinates of the ith landmark and the corresponding surface normal. For our mesh models, we take the triangle centers as landmarks. Furthermore, let $F(\boldsymbol{x})$ be the match function to be evaluated. Then, a target point search along a 1-dimensional profile of sampling points $\boldsymbol{x}_i^j = \boldsymbol{x}_i + j \cdot \delta \cdot \boldsymbol{n}_i$ (δ = sampling distance, $l \cdot \delta$ = search range) yields[1]:

$$\boldsymbol{x}_i^{\text{target}} = \operatorname{argmax}_{\{\boldsymbol{x}_i^j |\ j=-l,\ldots,+l\}} \left(F(\boldsymbol{x}_i^j) - D \cdot (\boldsymbol{x}_i^j - \boldsymbol{x}_i)^2 \right). \tag{2}$$

Here, the search can be biased to nearby target points by the heuristic distance penalty with weighting factor D.

To simulate the boundary detection during model adaptation per landmark, the given reference position $\boldsymbol{x}_i^{\text{ref}}$ of the landmark is deliberately displaced to some $\boldsymbol{x}_i$ and the target point is found according to (2) within the search profile around $\boldsymbol{x}_i$. In the simplest case, both the search and the displacements extend along the reference surface normal $\boldsymbol{n}_i^{\text{ref}}$. More realistic simulations include lateral displacements (parallel to the model surface) and also tilted normals $\boldsymbol{n}_i$ to vary the search direction. For match functions comparing gradient directions with surface normals, tilted $\boldsymbol{n}_i$ will also affect this comparison. Boundary detection errors are then measured as distances d between the detected target points and the given reference surface, or – computationally simpler – the tangent plane through $\boldsymbol{x}_i^{\text{ref}}$. This measurement is repeated for all simulated search profiles and for all training images. Finally, we calculate an *average error* for the tested match function F and the ith landmark as follows:

$$d_i(F) = \left(\frac{\sum d^n}{\sum 1} \right)^{1/n}. \tag{3}$$

Here, the sum runs over all training images and all simulated search profiles and d is the measured error per profile. The parameter n allows to put more emphasis on higher errors. We use $n = 2$ (resulting in a RMS error) to increase the robustness of the selected match functions against distant false attractors.

We note that local optimization using this performance measure is applicable to arbitrary functional forms of the match functions such as Mahalanobis distances or constrained edge detectors. Our experiments will use the latter, but this is no limitation to the method.

4. Experimental data and definitions

4.1. Introduction

This section presents the images from various modalities, mesh models and match functions, and the error measure that will be used in our experiments on different aspects of the Simulated Search. These will be reported in Section 5. Examples for the different images can be found in Fig. 6.

---

[1] The search along a 1-dimensional profile should be taken as example. Other sampling schemes could be used instead.

4.2. Image data

4.2.1. CT images

From a large database of retrospectively ECG-gated cardiac multi-slice CT images with large variations in contrast and image quality from 16-, 40-, and 64-slice scanners, 13 patients were arbitrarily selected with images from 1–3 different cardiac phases, amounting to 28 images overall. The mean in-plane voxel resolution is $0.48 \times 0.48\ \mathrm{mm}^2$ using a 512×512 matrix, and slice thickness varies from 0.67 to 3.0 mm. All images cover the complete heart.

We note that CT images have calibrated gray values defined on the Hounsfield scale. It will thus be plausible to use image features calculated from the original image gray values.

4.2.2. Non-contrasted SSFP MRI data

Using steady-state free-precession (SSFP = balanced TFE) MRI (Giorgi et al., 2002; Ozgun et al., 2005), 42 image volumes from different patients were acquired in various clinics with 1.5T scanners. Acquisition was performed without contrast administration at end diastole for coronary artery inspection. Images were reconstructed from multiple cardiac cycles using breathing motion compensation. In-plane voxel resolution is 0.5×0.5–$0.7 \times 0.7\ \mathrm{mm}^2$ using a 512×512 matrix, and slice distances vary from 0.7 to 0.9 mm. Some images do not cover the complete heart but show minor truncations (missing bottom or top axial slices).

For MRI, as opposed to CT, gray values are *not* calibrated. Even within one fixed protocol (like SSFP), gray values between different image volumes may exhibit some variable scaling and/or offset. We also note that the MRI images appear subjectively more noisy than the CT images.

4.2.3. Rotational X-ray angiography images

To assess the geometry of the left atrium and the pulmonary veins during interventions in the catheter laboratory (e.g., during atrial fibrillation ablation procedures), intra-procedurally acquired images may be segmented. Thirty-three contrast-enhanced rotational X-ray angiography image volumes (3D-RA, acquired during interventions at one clinical site) provided an even more challenging imaging modality and data base to test our training and segmentation approach. The image acquisition and reconstruction technology is explained in more detail in Manzke et al. (2006), Manzke et al. (in press). Image volumes consist of 256^3 voxels with isotropic voxel extensions of 0.43 mm.

Here, the images do not cover the complete heart but focus on the left atrium (LA) and the proximal parts of the pulmonary veins (PVs).

As compared to CT and MRI images, the 3D-RA images suffer from more artifacts and increased noise (image quality is substantially lower). As for MRI, gray values are not calibrated in 3D-RA.

4.3. Cardiac shape models and reference segmentations

For the whole heart scans from CT and MRI, our cardiac shape model is represented by a triangulated surface mesh with $T = 14771$ triangles and $V = 7286$ vertices (von Berg and Lorenz, 2005). This mesh encloses all four cardiac chambers, the left ventricular myocardium, and short trunks of the major vasculature (aorta, pulmonary artery and veins) as shown in Fig. 1.

For the 3D-RA images covering the left atrium (LA) and the pulmonary veins (PVs), a corresponding LAPV mesh (with trunks for the four standard PVs) with $V = 1330$ vertices and $T = 2717$ triangles was generated. This model is illustrated in Fig. 2.

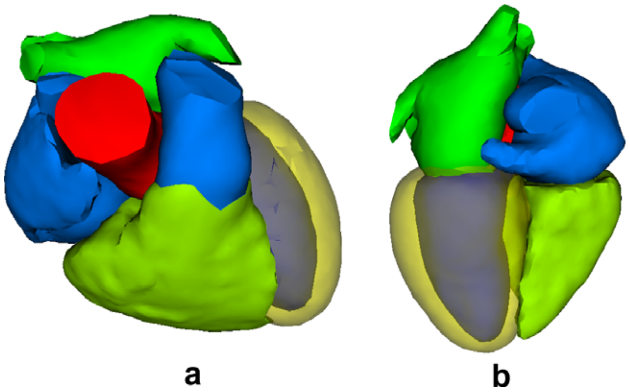


Fig. 1. Cardiac mesh with 14771 triangles, seen from top right and bottom. For the left ventricle, we model both the endo- and the epicardium (dark blue inside yellow). All other chambers are modeled by endocardial walls . Vessels are arbitrarily truncated.

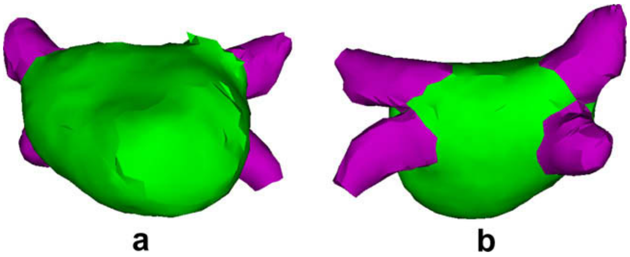


Fig. 2. LAPV mesh with 2717 triangles, seen from mitral annulus and from posterior (green = atrium, pink = PVs) (the atrial appendage is truncated, see top right in (a) close to the superior left PV).

Reference segmentations with identical mesh topology were generated by the following bootstrap procedure (see also Kaus et al., 2003; Ecabert et al., 2008): per modality, a few image volumes were manually annotated and the surface mesh was adapted to the classified regions. Boundaries for this first adaptation are thus defined by the volumetric classification. Then, a preliminary boundary model was trained for the corresponding gray value images. The resulting model was adapted to other images in a semi-automatic fashion, correcting the mesh surface in axial, coronal, and sagittal viewports (see also Ecabert et al., 2006; Ecabert et al., 2008). Training was repeated and further images were included in the semi-automatic segmentation until all images had been processed. Reference meshes were obtained after final corrections by technical and clinical experts. For the quantitative experiments reported below, new boundary and mean shape models have been trained that are independent of those used during the bootstrap procedure.

We note that manual corrections for the complete model surface are tedious and the resulting meshes are not everywhere perfect. For some cardiac regions, boundaries are partly ambiguous. Examples are the papillary muscles or cardiac valves which are invisible in part of the images. In the 3D-RA data, the mitral annulus was sometimes not well defined. Here, corresponding CT or MRI data served as additional knowledge source to adjust the mesh (Manzke et al., in press).

The resulting reference segmentations are used both to train the model geometry (mean mesh) and the tested boundary models, and to evaluate the segmentation performance in a cross-validation approach. We remark that minor imperfections of the reference segmentations are no problem for the Simulated Search which always measures the distance between detected target points and the reference surface instead of binary classification errors. It is thus sufficient to find the correct boundary in the vicinity of the reference boundary and no perfect match is needed.

4.4. Match function design and parameterization

In our images, boundaries are associated with intensity transitions, although sometimes weak or blurred. We thus start with the image gradient ∇I. This gradient is projected onto the triangle normal $\boldsymbol{n}_i$ to suppress edges that deviate strongly from the local surface orientation. Furthermore, to control the maximum feature response for metal implants or other artifacts, we use a heuristic damping of gradients exceeding a threshold $g_{\max}$ (Weese et al., 2001):

$$G^{\text{limit}}_{\text{proj}}(\boldsymbol{x}) = (\boldsymbol{n}_i \cdot \nabla I(\boldsymbol{x})) \cdot \frac{g_{\max} \cdot (g_{\max} + \|\nabla I(\boldsymbol{x})\|)}{g^2_{\max} + \|\nabla I(\boldsymbol{x})\|^2}. \tag{4}$$

If the transition across the boundary at a given landmark is always bright-to-dark or vice versa, the expected sign $s = \pm 1$ of $G^{\text{limit}}_{\text{proj}}(\boldsymbol{x})$ can be learned. The simple function

$$F(\boldsymbol{x}) = s \cdot G^{\text{limit}}_{\text{proj}}(\boldsymbol{x}) \tag{5}$$

can then discriminate between correctly oriented and inverted transitions. Sometimes (e.g., due to unpredictable local tissue like fat in MRI) the sign is less predictable, and an unsigned magnitude may replace (5):

$$F(\boldsymbol{x}) = \left\| G^{\text{limit}}_{\text{proj}}(\boldsymbol{x}) \right\|. \tag{6}$$

As discussed in Section 2.2, additional criteria may strongly improve the distinction between correct and false boundaries. To this end, we use some local image features Q_j and some corresponding acceptance intervals $[\text{Min}_j, \text{Max}_j]$. The set S of features Q_j along with their acceptance intervals are used to constrain the admissible edges. If any feature value violates its interval, the match value of the potential edge is set to zero (Peters et al., 2005):

$$F(\boldsymbol{x}) = \begin{cases} s \cdot G^{\text{limit}}_{\text{proj}}(\boldsymbol{x}) & \text{if } Q_j \in [\text{Min}_j, \text{Max}_j] \quad \forall Q_j \in S, \\ 0 & \text{otherwise}. \end{cases} \tag{7}$$

As in (6), $s \cdot G^{\text{limit}}_{\text{proj}}(\boldsymbol{x})$ can be replaced by $\|G^{\text{limit}}_{\text{proj}}(\boldsymbol{x})\|$. In our experiments, we exploit the following features Q_j:

- $I_{\text{in},K}$: gray value "inside" the mesh: in a profile along the inward triangle normal, gray values $I(\boldsymbol{x})$ are sampled. We always use a sampling step size of 1 mm. Taking the average over the first K sampling points yields the inside gray value.
- $I_{\text{out},K}$: gray value "outside" the mesh (analogous averaging over K points along the outward normal).
- I'_K : directed first derivative of the local gray values along the triangle normal. A linear fit is established through $I(\boldsymbol{x})$ which is sampled along a profile of K points on each side of the triangle. The fit's slope defines I'_K.
- $I_{\text{in/out},K,R}$: inside or outside gray values with additional averaging parallel to the surface to reduce the influence of image noise (similar to Brejl and Sonka (2000), but in 3D). We implement this averaging via a hexagonal ring of 6 additional sampling points around each of the K profile positions. R is the hexagon radius. We use $R = 3$ mm.
- $\Delta I_{K,R} = I_{\text{out},K,R} - I_{\text{in},K,R}$: difference of averaged gray values across the triangle. Again, we use $R = 3$ mm. Note that variations of the difference may be smaller than variations of in- or outside gray values.

To cope with uncalibrated gray values in MRI, most published segmentation algorithms include some normalization step in the match function design as explained in Section 2. From the known approaches, we decided to apply the following simple gray value calibration (see also Nyul and Udupa, 1999; Bosch et al., 2001): we compute the cumulative histogram of the voxel gray values in the complete image volume. To characterize the image's "relevant" gray value range we determine the L- and $(100 - L)$-percentile of the histogram. The interval between these two gray values is mapped linearly to the interval $[0,1]$. Values below the L- and above the $(100 - L)$-percentile are mapped to 0 and 1, respectively. We will denote the resulting calibrated gray values by $\widehat{I}^L$ and derived quantities by $\widehat{I}^L_{\text{in},K}$, $\widehat{I}^{L'}_K$, $\Delta\widehat{I}^L_{K,R}$, etc. Basically, this method implements a data-driven window and level adjustment to convert all images towards a similar appearance.

4.5. Training and assignment of match functions

The match functions introduced above contain various free parameters. Here, we describe how these are estimated from image data and how the resulting functions are assigned to the mesh triangles.

For a given functional form or "template" of the match functions, the traditional cluster-based training approach starts by clustering all triangles into groups of similar appearance, see Section 2.5. For our templates of the form (7), we use the features Q_j to define the image appearance. Applying a standard k-means clustering algorithm (Duda et al., 2001) we obtain a predefined number M of clusters. For each cluster, the distribution of the Q_j values in all training images at the reference positions is analyzed. Acceptance intervals $[\text{Min}_j, \text{Max}_j]$ are then created such that outliers are rejected. More specifically, we determine Min_j as the N-percentile and Max_j as $(100 - N)$-percentile of the Q_j distribution. This is done per cluster, resulting in M acceptance intervals. In addition, if the signed gradient (5) is used, the majority sign s is determined per cluster. The resulting match function parameters are assigned to all triangles of the respective cluster.

For local optimization via the Simulated Search, a variety of reasonable parameter settings is offered and the best parameterization per triangle will be selected. How reasonable parameters should be determined is arbitrary. As practical implementation for our experiments, we again apply the triangle clustering and the above rules to estimate $[\text{Min}_j, \text{Max}_j]$ and s. Here, however, we vary M and N to obtain multiple estimates for $[\text{Min}_j, \text{Max}_j]$. E.g., it is a priori unclear which number M of clusters will produce the best compromise between sufficient statistics and local specificity. Also, for the width of the acceptance intervals we have a tradeoff between rejecting too many true boundaries versus accepting too many false ones. Apart from the mere parameter optimization, locally useful features Q_j can be selected and harmful ones can be discarded if match functions using different templates or feature sets S are offered. E.g., features with different sampling ranges K can be tested and the set S can be enlarged or reduced, if this reduces the expected error. We thus propose to select the locally optimal match function from a large number of match function candidates. In the candidate list, we always offer a "Null" function which always gives a zero response. This match function will be selected in places where any other candidate would – on average – detect a misleading boundary.

Table 1 summarizes the training setups for different boundary models used in Section 5. Explanations for these setups will be given in the respective sections. In all experiments with the Simulated Search, the simulation covers 147 different poses for each triangle in each image (displacements $\leqslant 10$ mm both along triangle normal and laterally along three directions, plus tilting by 30° around three axes).

4.6. Error measure

In all experiments, segmentation errors are measured using the following symmetrized, constrained surface-to-surface distance: for each triangle center $\boldsymbol{c}_i$ of one mesh, we measure the distance

Table 1
Training setups for various boundary models *CT1–3DRA*. For the training approaches and parameters M (for clustering) and N (for $[\mathrm{Min}_j, \mathrm{Max}_j]$) see Section 4.5. For the Q_j used in the constraints see Section 4.4. (K = sampling range along triangle normal, R (fixed to 3 mm) = additional averaging parallel to surface, $\widehat{I}^L$ = calibrated gray values, I' = directed derivative, ΔI = difference across mesh.) Local optimization exploits various combinations S of the listed Q_j.

CT1	Global assignment to all triangles Match function: (5) with $s = -1$ (i.e., $-G_{\mathrm{proj}}^{\mathrm{limit}}$)
CT2	Cluster-based training and assignment to triangles Template: (7) with $s \cdot G_{\mathrm{proj}}^{\mathrm{limit}}$ (1 value for $g_{\max}$) Q_j: $I_{\mathrm{in},K}$ and $I_{\mathrm{out},K}$ ($K = 4$) $M = 10$; $N = 10$ (10 trained match functions)
CT3	Local optimization via Simulated Search Templates: (7) with $s \cdot G_{\mathrm{proj}}^{\mathrm{limit}}$ (1 value for $g_{\max}$) Q_j: $I_{\mathrm{in},K}$, $I_{\mathrm{out},K}$ ($K = 1,4$); I'_K ($K = 2,4$) $M = 5,10$; $N = 5,10$ (423 candidate functions)
CT4	Local optimization via Simulated Search Templates: (7) with $s \cdot G_{\mathrm{proj}}^{\mathrm{limit}}$ (1 value for $g_{\max}$) Q_j: $I_{\mathrm{in},K}$, $I_{\mathrm{out},K}$ ($K = 1,4$); I'_K ($K = 2,4$); $\widehat{I}^L_{\mathrm{in},K}$, $\widehat{I}^L_{\mathrm{out},K}$ ($K = 1,4$); $\widehat{I}^{L\prime}_K$ ($K = 2,4$); (L=2) $M = 5,10$; $N = 5,10$ (829 candidate functions)
MR1	Local optimization via Simulated Search Templates: (7) with $s \cdot G_{\mathrm{proj}}^{\mathrm{limit}}$ and $\lVert G_{\mathrm{proj}}^{\mathrm{limit}} \rVert$ (1 $g_{\max}$ value) Q_j: $\widehat{I}^L_{\mathrm{in},K,R}$, $\widehat{I}^L{}_{\mathrm{out},K,R}$, $\widehat{I}^{L\prime}_{K,R}$, $\Delta\widehat{I}^L_{K,R}$ ($K = 4$); ($L = 0,2,5$) $M = 5,10$; $N = 5,10$ (2709 candidate functions)
MR2	Local optimization via Simulated Search Templates: (7) with $\lVert G_{\mathrm{proj}}^{\mathrm{limit}} \rVert$ (2 values for $g_{\max}$) Q_j: $\widehat{I}^L_{\mathrm{in},K,R}$, $\widehat{I}^L{}_{\mathrm{out},K,R}$ ($K = 1,4$); $\Delta\widehat{I}^L_{K,R}$ ($K = 2,4$); ($L = 2$) $M = 5,10$; $N = 5,10$ (845 candidate functions)
MR3	Local optimization via Simulated Search Templates: (7) with $\lVert G_{\mathrm{proj}}^{\mathrm{limit}} \rVert$ (2 values for $g_{\max}$) Q_j: $I_{\mathrm{in},K,R}$, $I_{\mathrm{out},K,R}$ ($K = 1,4$); $\Delta I_{K,R}$ ($K = 2,4$); $\widehat{I}^L_{\mathrm{in},K,R}$, $\widehat{I}^L_{\mathrm{out},K,R}$ ($K = 1,4$); $\Delta\widehat{I}^L_{K,R}$ ($K = 2,4$); (L=2) $M = 5,10$; $N = 5,10$ (1689 candidate functions)
3D-RA	Local optimization via Simulated Search Templates: (7) with $s \cdot G_{\mathrm{proj}}^{\mathrm{limit}}$ and $\lVert G_{\mathrm{proj}}^{\mathrm{limit}} \rVert$ (4 $g_{\max}$ values) Q_j: $\widehat{I}^L_{\mathrm{in},K,R}$, $\widehat{I}^L{}_{\mathrm{out},K,R}$ ($K = 1,4$); $\Delta\widehat{I}^L_{K,R}$ ($K = 2,4$); ($L = 2$) $M = 5,10,20$; $N = 5,10$ (7857 candidate functions)

to a corresponding surface patch Π_i in the other mesh. This patch is constrained (1) by a geodesic radius $r = 10$ mm over the mesh surface around $\boldsymbol{c}_i$ in the other mesh and (2) by anatomical labels associated with all triangles. We average the distance from $\boldsymbol{c}_i$ to the closest point within Π_i from the automatic to the reference mesh and vice versa (these distances are not symmetric). This results in an error $\varepsilon_{i,n}$ per triangle i and per image I_n. Averaging over all images and over triangles from anatomical regions or the whole mesh, we can derive corresponding mean errors $\varepsilon_{\mathrm{region}}$ or $\varepsilon_{\mathrm{mesh}}$. We note that distal ends of our arbitrarily truncated vessels have no anatomically defined location along the vessel course. The measured errors $\varepsilon_{i,n}$ for distal triangles may thus diagnose displacements along the vessel course that do not relate to segmentation errors. We thus exclude contributions of these vessel-end regions from the error statistics. For the cardiac mesh from Fig. 1, 1035 of the 14771 triangles are excluded. For the LAPV mesh from Fig. 2, 255 of the 2717 triangles are excluded. Many of these triangles constitute artificial vessel caps.

5. Experiments

5.1. Overview

Experiments are designed to analyze different aspects of our approach. In Section 5.2, we report for the first time on a detailed analysis of improvements in terms of capture ranges and accuracies associated with optimized boundary detectors. In Section 5.3, we illustrate the selection behaviour of the Simulated Search. For new training tasks, we show how the method can help to find promising functional forms and parameter settings. In Section 5.4, finally, we show that heart segmentation in different modalities is feasible within the same algorithmic framework. Here, the Simulated Search is the key to train modality-specific boundary models.

5.2. Capture ranges and final accuracies

5.2.1. Introduction

In this section, we analyse capture ranges and final segmentation accuracies for boundary models obtained from traditional cluster-based training or from local optimization with the Simulated Search. Experiments are performed on the 28 cardiac CT image volumes from Section 4.2.1. Evaluation follows a leaving-one-patient-out approach, i.e., the model used to segment images from one patient is trained on the data excluding all cardiac phases from this patient.

Our study compares the first three boundary models from Table 1. As baseline, we evaluate the untrained edge detector *CT1* without additional features. Prior knowledge about the surface orientation yields $s = -1$ in (5). To reject false boundaries, constraints are added and match functions of the form (7) are trained. Training is either performed in the cluster-based scheme, yielding the boundary model *CT2*, or we optimize the match functions locally via the Simulated Search, resulting in *CT3*. In all models, $g_{\max}$ of $G_{\mathrm{proj}}^{\mathrm{limit}}$ is set to a fixed value that had been roughly optimized in previous experiments. Please note that the cluster-based training for *CT2* involves hard decisions about the match function design (i.e., we fix the set S for the constraints) and about the parameter estimation (M and N) The design and parameter estimation for *CT2* listed in Table 1 was chosen as most promising based on previous experiments.

5.2.2. Experimental approach

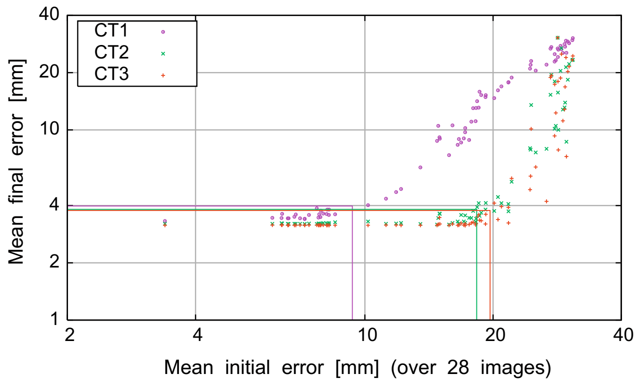
To define and analyze the capture range, we choose the following experimental approach: instead of using some manual or automatic initialization, we create systematically "perturbed" initializations with varying initial errors. We then run the segmentation using three algorithms with increasing degrees of freedom and analyze the final errors.

To simulate a large variety of initialization errors, we start from an optimal model pose that is achieved by registering the (leaving-one-patient-out) mean mesh to the reference mesh using one global similarity transformation. We then apply a series of pose perturbations to the registered mesh. Each such perturbation produces a more or less erroneous initialization. The applied pose perturbations are defined as single translations by 3–36 mm, or rotations by 10–60°, or scalings of the mesh by 60–140% around its center of gravity, or as "helical" operations combining translation and rotation. Translations and rotations use Cartesian ($\pm x, \pm y, \pm z$) or diagonal directions and axes. In the experiments below, we apply different ranges of pose perturbation errors to explore meaningful situations for each tested algorithm. The simulation covers one unperturbed and 80–120 perturbed initializations per algorithm. The experimental workflow is summarized in Fig. 3.

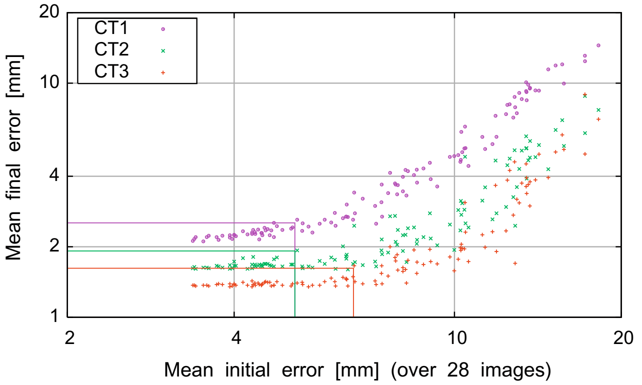
The algorithms to test the different boundary models have been published in Ecabert et al. (2008) and are described in detail in Appendix A.2, A.3, A.4. The first two algorithms use a parametric shape model while the third one allows for local shape deformations. In brief, they work as follows:

1. Register model to reference mesh	
2. Apply pose perturbation	$\rightarrow \varepsilon_{\text{mesh}}^{\text{init}}$
3. Apply selected adaptation algorithm	$\rightarrow \varepsilon_{\text{mesh}}^{\text{final}}$

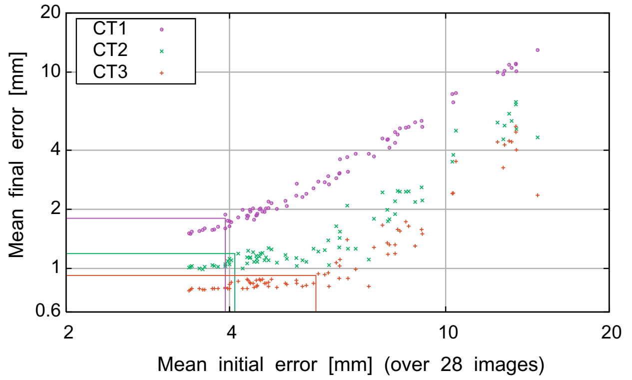
Fig. 3. Experimental workflow. Errors are measured for the perturbed initialization ($\varepsilon_{\text{mesh}}^{\text{init}}$) and after adaptation with the tested algorithm ($\varepsilon_{\text{mesh}}^{\text{final}}$). Steps 1 and 2 simulate an imperfect automatic model initialization and the experiments cover a large variety of pose perturbations.



(a) Global pose adaptation.



(b) Piecewise affine adaptation.



(c) Deformable adaptation.

Fig. 4. Scatter plots of final errors (after adaptation) versus initial errors (after pose perturbation) for the boundary models *CT1* (pink ⊙), *CT2* (green ×), and *CT3* (red +). Here, each data point represents the mean segmentation error over the mesh and over all 28 images for one fixed pose perturbation. Errors are plotted on a log–log scale to cover the wide range of variations. The colored lines indicate one option to define thresholds on $\varepsilon_{\text{mesh}}^{\text{final}}$ and the resulting ranges of robust segmentation.

- *Global pose adaptation:* Here, the shape model is treated as a rigid object that can be translated and rotated. Furthermore, isotropic scaling is allowed. Adaptation thus optimizes one global similarity transformation for the complete mesh with respect to a so-called external energy that penalizes deviations between the mesh and the detected boundaries.
- *Piecewise affine adaptation:* Here, the cardiac mesh is divided into five connected parts (four chambers + myocardium, the vessel trunks are assigned to the connected chambers). The five sub-meshes are treated with independent affine transformations. These are interpolated for smooth mesh transitions in the connecting mesh regions. Now, the five affine transformations are optimized with respect to the external energy.
- *Deformable model adaptation:* To allow for local deformations towards the individual organ shape, shape-constrained deformable models as introduced in Weese et al. (2001) are used. Using the piecewise affine transformations, a space of cost-free shape variations is constructed. Deviations from this shape space are penalized using a so-called internal energy. All vertex coordinates are now optimized with respect to a weighted combination of internal and external energy.

In our experiments, the target point search (2) always uses a sampling distance of $\delta = 1$ mm. For the deformable model adaptation, the relative weight for internal versus external energy has been optimized in previous experiments (Ecabert et al., 2008). Different search ranges are used for parametric and deformable adaptation. The latter (and the Simulated Search for *CT3*) uses a search range of $l \cdot \delta = 10$ mm. For both versions of the parametric adaptation, we extend the search range to 30 mm. We terminate the iterative adaptation if maximum vertex displacement over five iterations falls below 2.5 mm or after a predetermined maximum number of iterations (20 for deformable and 10 for parametric adaptation).

5.2.3. Qualitative results

Scatter plots of $\varepsilon_{\text{mesh}}^{\text{final}}$ versus $\varepsilon_{\text{mesh}}^{\text{init}}$ allow to study the influence of the initialization quality. Fig. 4 presents the scatter plots for the three tested boundary models and the different adaptation algorithms. Here, $\varepsilon_{\text{mesh}}^{\text{final}}$ and $\varepsilon_{\text{mesh}}^{\text{init}}$ are averaged over the whole mesh and over all 28 images. Per adaptation algorithm, each initial pose perturbation thus leads to one data point per boundary model.

In the scatter plots of Fig. 4, we observe the following:

- Over some range of initial errors, we observe small fluctuations of the final errors, i.e., segmentation is robust. For initial errors exceeding some threshold, however, the segmentation clearly deteriorates. This is most evident for the global pose adaptation in Fig. 4a. Here, with *CT2* and *CT3*, the correct pose is mostly recovered for initialization errors of up to ca. 20 mm. Thereafter, adaptation degrades rapidly. For the simple edge detector *CT1*, degradation already starts around $\varepsilon_{\text{mesh}}^{\text{init}} = 10$ mm.
- For the piecewise affine and the deformable adaptation in Fig. 4b and c, the final accuracy depends clearly on the boundary model. The locally optimized functions of *CT3* yield the best results. Stable results are now restricted to smaller initial errors (for all boundary models). The plateau-like distribution of final errors is now less evident than for the global pose adaptation.

5.2.4. Quantitative results

These qualitative observations motivate a more quantitative definition of three performance measures. We first quantify the range of robust segmentation per model, i.e., the extent of the horizontal plateaus in Fig. 4. This measure ignores the fact that these plateaus occur at different final accuracies. The capture range then quantifies the allowable initialization error to reach a fixed final accuracy. Finally, we compare the final accuracies for moderate initialization errors.

- *Range of robust segmentation:* To characterize stable performance per boundary model we allow some relative degradation of $\varepsilon^{\text{final}}_{\text{mesh}}$ as compared to the best result with the tested model. We thus define a threshold $\varepsilon^{\text{final}}_{\text{max}}$ per model. The range of robust segmentation is then defined as maximum $\varepsilon^{\text{init}}_{\text{mesh}}$ among all tested initializations for which $\varepsilon^{\text{final}}_{\text{mesh}} < \varepsilon^{\text{final}}_{\text{max}}$. We explain this scheme for the global pose adaptation algorithm, referring to Fig. 4a. The minimum $\varepsilon^{\text{final}}_{\text{mesh}}$ observed for model *CT1* is 3.3 mm. Allowing an increase of 20% yields $\varepsilon^{\text{final}}_{\text{max}} = 4.0$ mm. Allowing the same relative increase of 20% for the slightly better models *CT2* and *CT3* yields $\varepsilon^{\text{final}}_{\text{max}} = 3.8$ mm. This construction is illustrated with the colored lines for all adaptation algorithms in Fig. 4. Horizontal lines depict the thresholds $\varepsilon^{\text{final}}_{\text{max}}$ and vertical lines depict the resulting ranges of robust segmentation. Table 2 summarizes these ranges numerically.
- *Capture range:* To assess the range of allowable initialization errors to reach a fixed accuracy we define a model-independent threshold $\varepsilon^{\text{final}}_{\text{max}}$ per adaptation algorithm. The capture range is then defined as maximum $\varepsilon^{\text{init}}_{\text{mesh}}$ for which $\varepsilon^{\text{final}}_{\text{mesh}} < \varepsilon^{\text{final}}_{\text{max}}$. The values for $\varepsilon^{\text{final}}_{\text{max}}$ are chosen between the horizontal lines of the two models *CT2* and *CT3* in Fig. 4. Table 3 lists the chosen $\varepsilon^{\text{final}}_{\text{max}}$ and the resulting capture ranges.
- *Final accuracy:* To characterize the segmentation quality for moderate initialization errors we average the final errors $\varepsilon^{\text{final}}_{\text{mesh}}$ over several initializations with errors below some threshold $\varepsilon^{\text{init}}_{\text{max}}$. The values for $\varepsilon^{\text{init}}_{\text{max}}$ are rounded from the capture ranges of *CT2*. Table 4 lists these $\varepsilon^{\text{init}}_{\text{max}}$ values and the resulting final errors per boundary model.

5.2.5. Discussion

The scatter plots in Fig. 4 and the quantitative results from Tables 2–4 show clearly improved accuracies over increasing capture ranges as we go from *CT1* to *CT3*. Applying constraints on the edge detectors in *CT2* is particularly helpful to increase the capture range during global pose adaptation (Tables 2 and 3). For the piecewise affine and deformable adaptation, the local optimization of the boundary model in *CT3* improves both the capture range and the resulting accuracy (Tables 2–4).

Table 2
Ranges of robust segmentation [in mm] for model dependent $\varepsilon^{\text{final}}_{\text{max}}$ allowing 20% relative degradation per boundary model.

Adaptation	*CT1*	*CT2*	*CT3*
Global pose	9.3	18.3	19.7
Piecewise affine	5.2	5.2	6.6
Deformable	3.9	4.1	5.8

Table 3
Capture ranges [in mm] using one fixed $\varepsilon^{\text{final}}_{\text{max}}$ for all models.

Adaptation	$\varepsilon^{\text{final}}_{\text{max}}$ (mm)	*CT1*	*CT2*	*CT3*
Global pose	3.8	7.6	18.3	19.7
Piecewise affine	1.8	–	4.3	7.6
Deformable	1.1	–	4.0	6.4

Table 4
Final errors (mm) averaged over initializations with errors $\leqslant \varepsilon^{\text{init}}_{\text{max}}$.

Adaptation	$\varepsilon^{\text{init}}_{\text{max}}$ (mm)	*CT1*	*CT2*	*CT3*
Global pose	18.0	5.78	3.27	3.16
Piecewise affine	4.0	2.09	1.63	1.33
Deformable	4.0	1.45	1.02	0.76

The results also show that the three adaptation algorithms have complementary strengths and weaknesses. The global pose adaptation is most robust due to the global shape constraint. Local misdetections of false boundaries do not confound the adaptation as long as sufficiently many correct boundaries are detected. Piecewise affine and deformable adaptation are less robust since shape constraints operate more locally. However, the increased degrees of freedom for the surface shape enable an improved accuracy.

5.3. Feature selection and distribution

In this section, we illustrate how the Simulated Search can be used to address a new segmentation task. Due to changed image characteristics, it is necessary to test available templates for the match functions, introduce new ones, or find suitable values for specific parameters. In principle, this could be done in a single extensive training with a comprehensive set of templates and parameters. As typical training times for the setups described here are in the order of days, an extensive training can become too time consuming. In this case, the Simulated Search can also be used to find classes of meaningful features and suitable values for specific parameters before the final training. The specific example we discuss stems from adapting the heart segmentation to a new imaging modality or imaging protocol.

Going from cardiac CT segmentation to the uncalibrated SSFP MRI images from Section 4.2.2, we introduce the calibration scheme denoted with $\widehat{I}^{L}$ in Section 4.4. To find a promising choice for the free parameter L used in this calibration, we performed one initial (not cross-validated) training offering three different values for L. In addition, we offered to use the signed or unsigned edge detector in (7) and offered to use the gray value difference across the mesh in the constraints. Here, the Simulated Search provides a performance measure to compare various settings of less intuitive parameters such as L or to compare different match function templates in order to find the most promising ingredients for the final model training. The setup for the initial training is described as *MR1* in Table 1. The resulting selection rates for different subsets of function candidates are listed in Table 5. These rates are defined by the percentage of triangles using the specified template or a feature Q_j from the specified class. E.g., 56% of all triangles select a function using features with calibration by $L = 2$. As compared to $L = 0$ and $L = 5$, this calibration seems best for this task if a single value alone should be selected. Furthermore, using (7) with $\|G^{\text{limit}}_{\text{proj}}\|$ instead of $s \cdot G^{\text{limit}}_{\text{proj}}$ seems preferable. These findings lead to the training setup *MR2* of Table 1 for the segmentation results presented in Section 5.4.

As stated above, we had no intuitive feeling for the best calibration parameter L. The feature types Q_j that are used to constrain

Table 5
Selection rates for various match functions for setup *MR1*.

F using ...	Selection (%)
(7) with $s \cdot G^{\text{limit}}_{\text{proj}}$	29.6
(7) with $\|G^{\text{limit}}_{\text{proj}}\|$	65.9
Q_j with $L = 0$	21.7
Q_j with $L = 2$	56.1
Q_j with $L = 5$	17.2
$Q_j = \widehat{I}^{L}_{\text{in},K,R}$	62.5
$Q_j = \widehat{I}^{L}_{\text{out},K,R}$	66.5
$Q_j = \widehat{I}^{L'}_{K,R}$	37.7
$Q_j = \Delta\widehat{I}^{L}_{K,R}$	40.3
“Null” function	4.5

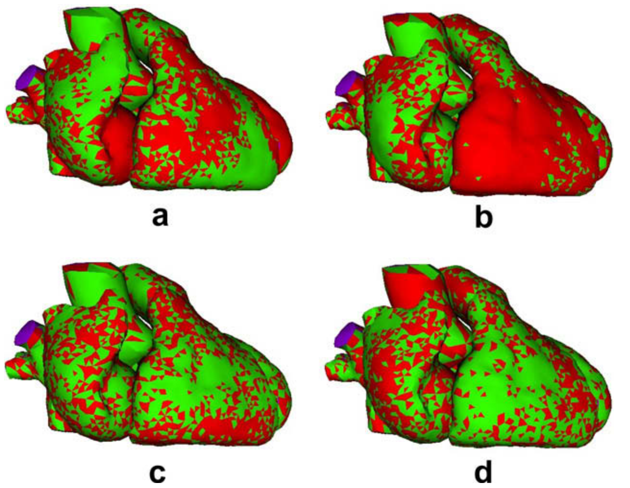


Fig. 5. Distribution of match functions for setup *MR1* using the following features Q_j: (a) $\widehat{I}_{\text{in}}$, (b) $\widehat{I}_{\text{out}}$, (c) $\widehat{I}'$, (d) $\Delta\widehat{I}$ (parameters L, K, R omitted). Red triangles indicate a selection of the respective feature Q_j while the green triangles do not use Q_j. The pictures illustrate the right chambers (left = atrium with the appendage that partly hides the aorta, right = ventricle) and parts of the large vessels. The pink triangles on the very left indicate that the "Null" function has been automatically selected for the cap of the truncated right upper pulmonary vein.

the admissible edges, however, have an intuitive interpretation: e.g., $\widehat{I}^{L}_{\text{in},K,R}$ with any value for L (right part of Table 5) checks the inside gray value. Fig. 5 shows how the selected features are distributed over the mesh surface. We first observe that the "Null" function is automatically assigned for most triangles of the arbitrarily truncated vessel caps. In Fig. 5, this can be seen for the right upper pulmonary vein. This selection makes sense since no edges should be detected. Surprisingly, most other results are not very intuitive. Looking at the selection for features $Q_j = \widehat{I}_{\text{in}}, \widehat{I}_{\text{out}}, \widehat{I}'$, and $\Delta\widehat{I}$, we find no evident pattern. E.g., it is not clear why the inside gray value $\widehat{I}_{\text{in}}$ is rarely selected for the right atrial appendage or the ascending aorta or near the right ventricular apex Locally, other features seem to yield a better performance such as $\Delta\widehat{I}$ for a large part of the aorta or the posterior part of the right atrium.

For comparison, we also report the selection rates for features using calibrated or uncalibrated gray values for CT and MRI. For CT, the setup *CT3* was modified into *CT4* to include calibrated features. For MRI, the setup *MR2* was modified into *MR3* to include uncalibrated features. The results are listed in Table 6. They confirm the expectations that the Hounsfield scale in CT delivers calibrated gray values that best characterize the different tissues while MRI clearly benefits from image calibration.

Overall, the results indicate that the Simulated Search helps to select classes of meaningful features and to identify suitable parameter settings. Locally, the distribution of the most preferable feature combination is less intuitive and the statistical training selects match functions that would be hard to assign by mere intuition.

5.4. Multi-modal heart segmentation

5.4.1. Introduction

Fully automatic whole heart segmentation in CT images has been presented in Ecabert et al. (2008). Section 5.4.2 briefly summarizes the results. Here, model adaptation cascades the three algorithms introduced in Section 5.2.2. To automatically initialize the mesh in a new image, we coarsely localize the heart using a 3D version of the Generalized Hough Transform (GHT) (see also Appendix A.5). Sections 5.4.3 and 5.4.4 show that the same framework is equally applicable for cardiac segmentation tasks in other imaging modalities. The Simulated Search is a key to easily train good boundary models from given training images and corresponding reference segmentations. In addition, the mean shape as well as the GHT parameters are re-trained. Minor adjustments of the adaptation chain to cope with particular properties per application will be explained below. Processing times on a standard PC are in the order of 10–30 s. Section 5.4.5 compares results for CT and MRI with the literature.

Table 6
Selection rates for using calibrated versus original gray values.

F using ...	CT (%)	MRI (%)
Q_j with calibrated gray values	23.1	90.2
Q_j with original gray values	72.6	4.8
"Null" function	4.1	4.8

5.4.2. CT results

For the CT images from Section 4.2.1, experiments are carried out in the leaving-one-patient-out approach already explained in Section 5.2.1. Now, the leaving-one-patient-out training includes the GHT parameters. We localize the heart via the GHT and then perform the global pose, the piecewise affine, and finally the deformable adaptation using the match functions *CT3* from Table 1. Table 7 lists the results (see Ecabert et al., 2008). Included is the distribution of triangle-specific errors ε_i over the mesh surface ($\varepsilon_{i,n}$ averaged over the 28 images n). This distribution indicates whether some mesh regions have systematically high errors. The standard deviations specified after the mean errors in Table 7 cover the variations of $\{\varepsilon_{i,n}\}$ both across the triangles i and across the images n. These numbers appear larger than expected and exceed the mean errors. This is due to the fact that errors across the triangles per image are not normally distributed and some few high errors result in a high numeric variance. Generally, we find that more than 90% of the errors are $< 2 \cdot \varepsilon_{\text{mean}}$. For variations of $\varepsilon_{\text{mean},n}$ across the images n as typically reported in the literature, see Table 10 in Section 5.4.5. Fig. 6a–d shows examples of segmentation results.

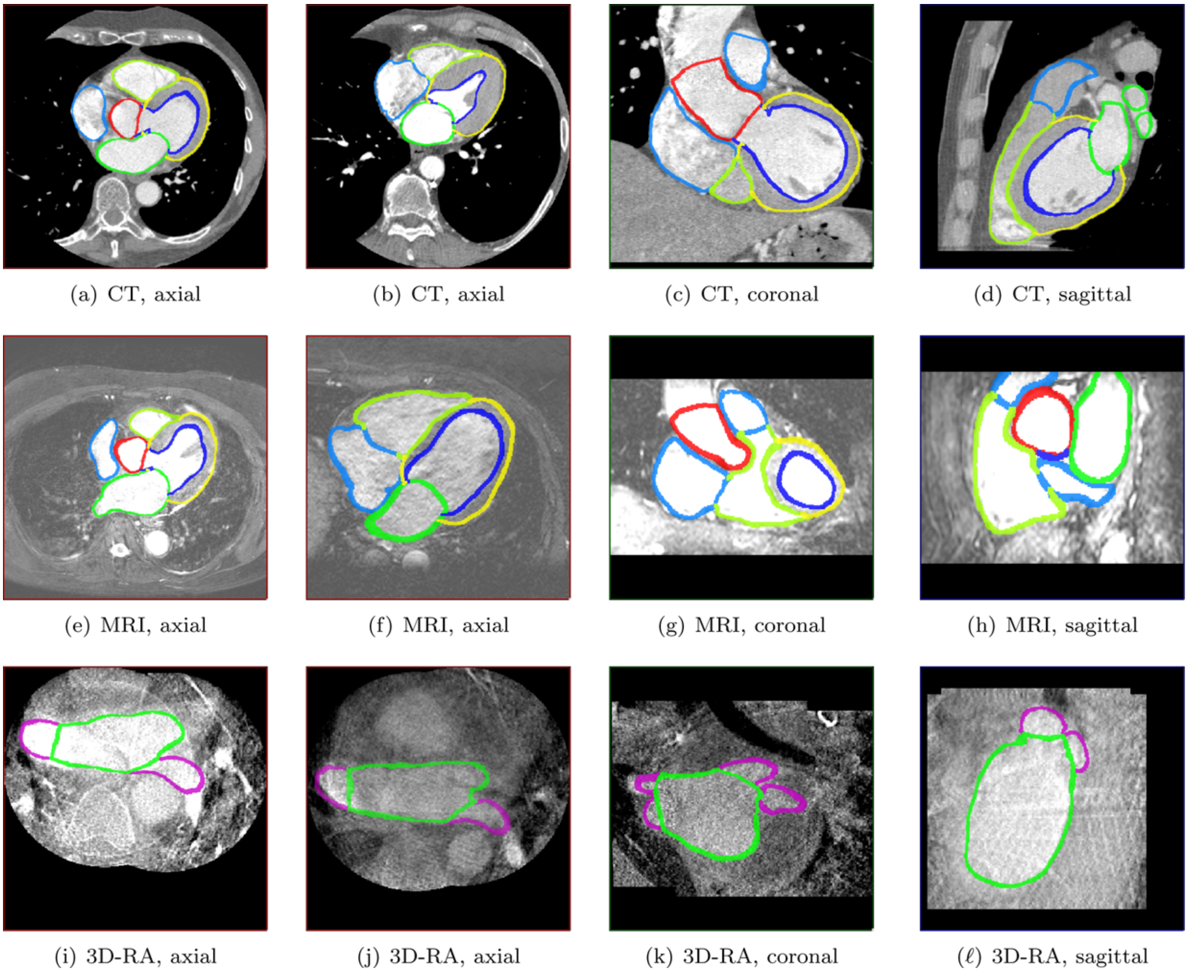
5.4.3. MRI results

For the MRI data from Section 4.2.2, the GHT parameters were adapted to localize the blood pool instead of the outer heart contours (see also Peters et al., 2007). During parametric adaptation, an additional step using a single affine transformation is performed after the global pose and before the piecewise affine adaptation. This step improves the final accuracies.

Match functions are trained as described in Table 1 under *MR2*. Training and evaluation are performed using 4-fold cross-validation. The 42 image volumes (all from different patients) are divided

Table 7
Mean errors and distribution of ε_i for the fully automatic segmentation of 28 CT images using the boundary model *CT3* (leaving-one-patient-out evaluation). The standard deviation specified after the mean error covers the variations of $\{\varepsilon_{i,n}\}$ both across the triangles i and across the images n. For variations across the images n only, see Table 10 in Section 5.4.5.

	Mean ± SDev	<1 mm (%)	1–2 mm (%)	⩾2 mm (%)
Whole mesh	0.82 ± 1.00	73.7	25.9	0.4
Left atrium	0.71 ± 0.88	84.8	12.2	3.0
LV endocardium	0.98 ± 1.32	54.5	43.6	1.9
LV epicardium	0.82 ± 1.07	74.2	25.7	0.1
Right atrium	0.89 ± 0.96	65.1	34.8	0.1
Right ventricle	0.84 ± 0.94	72.9	27.0	0.1
Aorta trunk	0.74 ± 0.89	85.6	14.4	0.0
Pulm. artery	0.83 ± 1.05	77.1	22.0	0.9


(a) CT, axial (b) CT, axial (c) CT, coronal (d) CT, sagittal

(e) MRI, axial (f) MRI, axial (g) MRI, coronal (h) MRI, sagittal

(i) 3D-RA, axial (j) 3D-RA, axial (k) 3D-RA, coronal (ℓ) 3D-RA, sagittal

Fig. 6. Fully automatic segmentation results for all discussed imaging modalities (for 12 different patients, using the respective cross-validation models, some images zoomed). To illustrate contrast variations, window and level settings were fixed for CT and for MRI. For 3D-RA, window and level had to be adjusted per image. In some cases with unclear boundaries (especially for 3D-RA), the correct anatomy can often be assessed only in 3D, and inspection in all directions (axial, coronal, sagittal) and through many slices is needed to appreciate the segmentation.

into four sets of 10 or 11 volumes each, and evaluation is performed per set with a model trained on the other three sets.

Table 8 summarizes the errors. These are quantitatively close to those reported for CT in Table 7. Examples for the segmentation are shown in Fig. 6e–h.

Table 8
Mean errors (± standard deviation of $\{\varepsilon_{i,n}\}$) and distribution of ε_i for 42 steady-state free-precession MRI volumes (fourfold cross-validation).

	Mean ± SDev	<1 mm (%)	1–2 mm (%)	⩾2 mm (%)
Whole mesh	0.76 ± 1.08	76.6	22.7	0.7
Left atrium	0.72 ± 1.14	80.0	18.1	1.9
LV endocardium	0.69 ± 1.13	83.2	16.8	0.0
LV epicardium	0.83 ± 1.17	68.3	30.5	1.2
Right atrium	0.63 ± 0.95	87.7	12.3	0.0
Right ventricle	0.74 ± 0.96	81.1	18.9	0.0
Aorta trunk	0.60 ± 0.70	98.5	1.5	0.0
Pulm. artery	0.73 ± 1.11	78.4	21.2	0.4

5.4.4. 3D-RA results

For the 3D-RA data from Section 4.2.3 covering the left atrium and the pulmonary veins, the LAPV model from Fig. 2 was used. Here, the parametric adaptation retains the single affine step explained in Section 5.4.3. The segmentation algorithm is slightly modified as described in Peters et al. (2008): to cope with the less predictable geometry of the vessel course, the pulmonary veins are subdivided into proximal, medial, and distal segments. The atrium plus the proximal segments share one affine transformation and each medial and distal segment obtains an own affine transformation to allow for local re-orientation. During parametric and initial deformable adaptation, medial and distal segments are excluded from the external energy. After adapting the atrial mesh and the proximal parts of the PVs to the image, the local orientation per vessel is estimated and the medial and distal PV segments are re-oriented. Adaptation continues, now including the medial segments of the PVs into the external energy. From the adapted medial segments, the orientation of the distal segments is finally estimated and adjusted and a final adaptation is performed for medial and distal PV segments.

The GHT was adapted to the new model and boundary models were trained according to 3D-RA from Table 1. This training setup again includes image calibration because a rotational angiography protocol without gray value calibration was used and the contrast agent density in the left atrium was varying across patients. The calibration scheme introduced in Section 4.4 for MRI is used here as well to compensate offsets and scalings in the gray values. Experiments use 11-fold cross-validation, i.e., leaving-3-out models are trained to segment the three left-out images.

Table 9
Mean errors ($\pm$ standard deviation of $\{\varepsilon_{i,n}\}$) and distribution of ε_i for 33 3D-RA volumes (11-fold cross-validation).

	Mean ± SDev	<1 mm (%)	1–2 mm (%)	⩾2 mm (%)
Whole mesh	1.31 ± 1.50	48.6	34.9	16.5
Left atrium	1.50 ± 1.70	37.3	38.9	23.8
Inf. left PV	0.86 ± 0.95	79.8	20.2	0.0
Inf. right PV	0.92 ± 0.89	72.3	27.7	0.0
Sup. left PV	0.84 ± 0.81	80.3	19.7	0.0
Sup. right PV	1.25 ± 1.11	37.2	51.8	11.0

Table 9 summarizes the results for this segmentation task. As compared to the accuracies achieved on CT and MRI data we observe larger errors. This, however, was expected since the boundaries in the 3D-RA images are more ambiguous (even for human experts). Especially, the anterior walls of the left atrium lack contrast and the transition to the left ventricle is not clearly defined in the images. For a deeper discussion and comparisons of 3D-RA segmentation versus meshes obtained from pre-procedurally acquired CT or MRI data, see Manzke et al. (in press). Segmentation examples are shown in Fig. 6i–l.

5.4.5. Comparison with literature

Early work on cardiac image segmentation focussed exclusively on the left ventricular endo- and epicardial walls. Atlas-based methods and shape models have been applied to 2D, 3D, and 4D (3D + time) data sets from different imaging modalities. For reviews on 3D left ventricle models and segmentation see, e.g., Frangi et al. (2001), Kaus et al. (2004), van Assen et al. (2006). Surface-to-surface errors for MR images reported therein from various research groups range from 1.9 to 2.9 mm for endo- and epicardial walls. Epicardial errors tend to be higher than endocardial errors.

In Mitchell et al. (2001), right and left ventricular boundaries are segmented in selected 2D short-axis slices from 60 MR images. Here, Active Appearance Models and Active Shape Models are cascaded and image appearance is characterized by Mahalanobis distances. In Lötjönen et al. (2004), a first comprehensive 3D cardiac model with all four chambers plus epicardium was presented. A probabilistic atlas is registered to 25 short- and long-axis cardiac MR images. In Lorenz and von Berg (2006), the mesh model used in our work is adapted to 10 from the 28 cases of our CT data set. The used gray value constrained edge detectors are not optimized via the Simulated Search. In Zheng et al. (2008), four cardiac chamber models plus an LV epicardial model are adapted to 323 CT images. Automatic object and pose detection uses a new approach (marginal space learning) and the final adaptation uses a PDM shape model. Boundary detection is performed as explained in Section 2.4. Atlas-based methods are recently also applied to whole heart segmentation of 3D MR images. In Wierzbicki et al. (2008), models for five cardiac substructures are registered to 10 MR data sets with thick slices (6 mm). Zhuang et al. (2008) report on an atlas-based whole heart segmentation in 19 3D MR images with 2 mm voxel size. Finally, a comparison of results using one single algorithm for 25 CT and 15 MR images is presented in van Assen et al. (2008). Although only the left ventricle is segmented in thick short-axis slices and errors are measured for contours per slice, we include the results here to cover another multi-modal approach.

Table 10 summarizes the segmentation errors of our approach after initialization with the GHT, after piecewise affine model adaptation, and after final segmentation for CT and MR images together with comparable results reported in the literature. An important difference between the approaches is that apart from Lorenz and von Berg (2006), Zhuang et al. (2008) and our approach, all whole heart segmentations use five distinct meshes. Initial adaptation of these individual meshes involves one registration per mesh which is similar to our piecewise affine model adaptation (see Section 5.2.2 and Appendix A.3 and A.4). The respective segmentation errors are, therefore, listed in the column "Piecewise affine" of Table 10. According to Table 10, our approach leads to the best segmentation results with the smallest segmentation errors. We note, however, that quantitative comparisons are difficult since data sets differ not only in contents (patients, cardiac phases, applied contrast protocols) but also in image quality and resolution. Also, the cardiac models and the error metrics are not fully equivalent. Finally, the process of ground truth creation may introduce a bias if the method being tested is involved in the semi-automatic definition of reference segmentations. Even after expert corrections, some acceptable mesh regions may remain unchanged and may result in too optimistic error numbers. This was reported in Lorenz and von Berg (2006) for a second test set where manual corrections were less strict than for the references used in our experiments.

6. Conclusion

Robust and accurate boundary detection for complex anatomical models can be achieved by locally optimized match functions. A dedicated framework using the Simulated Search has been introduced to estimate the performance for any proposed match function. This performance measure evaluates the geometric distance of detected target points from reference boundaries. The measure operates locally per model landmark and the resulting simulated

Table 10
Errors at various segmentation stages for CT and MR image segmentation as far as they are reported in the literature. Mean surface errors are given in mm. In parenthesis, standard deviations of mean errors across the image volumes are specified. These exclude the error variations within each image and are thus smaller than the SDev values reported in Tables 7–9. The Initialization column lists 'A' for automatic and 'M' for manual methods. The Regions column lists 'WH' for whole heart segmentation, 'LV' and 'RV' are left and right ventricles. Remarks: (a) Error ranges describe different images. (b) Error ranges cover the different anatomical regions. (c) Errors describe the average for the whole heart. For our method, the breakdown to single regions can be found in Tables 7 and 8. (d) RMS instead of mean errors are reported. (e) Errors are measured in 2D per slice. (5) Five distinct meshes are used.

Reference	Initialization	Piecewise affine	Final results	Modality	Regions	Remarks
van Assen et al. (2008)	M		1.02–2.97	CT	LV	(a,e)
Zheng et al. (2008)	A	2.51–3.17 (0.75–1.10)	1.13–1.57 (0.38–0.55)	CT	WH	(b) (5)
Lorenz and von Berg (2006)	M 2.87		1.11	CT	WH	(c)
Our approach	A 8.14 (3.65)	1.30 (0.21)	0.82 (0.24)	CT	WH	(c)
van Assen et al. (2008)	M		1.27–2.05	MR	LV	(a,e)
Mitchell et al. (2001)	A		1.71–2.46 (0.82–1.39)	MR	LV+RV	(b,d,e)
Lötjönen et al. (2004)	A		2.53 (0.70)	MR	WH	(c) (5)
Wierzbicki et al. (2008)	M	3.7 (0.7)	3.0 (0.5)	MR	WH	(c,d) (5)
Zhuang et al. (2008)	A	2.8 (0.40)	1.8 (0.42)	MR	WH	(c)
Our approach	A 8.38 (2.98)	1.53 (0.32)	0.76 (0.30)	MR	WH	(c)

errors are decoupled from functions assigned to neighboring landmarks. This measure allows local optimizations that are not practically feasible by brute-force testing of different globally trained boundary models.

For the example of fully automated cardiac segmentation, the advantages of the new training scheme are demonstrated. For CT images, the influence of increasing initialization errors on the resulting segmentation is analyzed and the new optimization scheme is compared to a traditional training approach. The local optimization of the boundary model via the Simulated Search clearly extends the capture range and at the same time improves the final accuracy. Examples illustrate that the Simulated Search can be used to identify suitable classes of features when addressing a new segmentation task. The Simulated Search also allows to easily optimize models for new imaging modalities and imaging protocols.

Multi-modal image segmentation using the same algorithm and the same framework to design boundary detection models is feasible as demonstrated for data bases from CT, MRI, and 3D rotational X-ray angiography. Starting from our initial experience with CT images, only minor modifications in the training setup including image calibration were needed. The Simulated Search automatically selected suitable parameter settings for the new imaging modalities. For all evaluated applications, fully automatic segmentation is achieved. This illustrates the robustness of the underlying boundary detection and the optimization scheme.

Acknowledgments

The authors would like to thank colleagues from Philips Research North America and from Philips Healthcare working for CT, Healthcare Informatics, X-ray, and Electrophysiology for providing clinical images, reference segmentations, and continuous feedback on our cardiac segmentation system. We also thank Cristian Lorenz and Jens von Berg from Philips Research Hamburg for the mesh models used throughout this paper.

Appendix A. Segmentation algorithms

A.1. Introduction

For completeness, this appendix defines the algorithms from Ecabert et al. (2008) that are used in the experiments of this paper. Our segmentation approach is based on a triangulated surface mesh with T triangles and V vertices. The mesh topology is fixed, i.e., no triangles or vertices are added or removed during adaptation. This property allows to attach dedicated information like anatomical labels or locally optimized match functions F_i to each triangle i.

We use three algorithmic variants for mesh adaptation. The first two use a parametric shape model while the third allows for local shape deformations. All algorithms involve the same external energy to attract the mesh towards detected boundaries. First, target points are searched along profiles $\{\boldsymbol{x}_i^j\}$ perpendicular to each triangle according to (2). The external energy then penalizes the squared distance of each adapting triangle center $\boldsymbol{c}_i$ from the plane tangent to the detected image boundary:

$$E_{\text{ext}} = \sum_{i=1}^{T} f_i \cdot \left(\frac{\nabla I(\boldsymbol{x}_i^{\text{target}})}{\|\nabla I(\boldsymbol{x}_i^{\text{target}})\|} \cdot \left(\boldsymbol{c}_i - \boldsymbol{x}_i^{\text{target}}\right) \right)^2. \tag{A.1}$$

Projecting $(\boldsymbol{c}_i - \boldsymbol{x}_i^{\text{target}})$ onto the direction of the image gradient ∇I allows cost-free lateral “sliding” of the triangles on the image boundaries. The influence of unclear boundaries is reduced by coupling the factors f_i with the “match value” in (2) – for negative values, f_i is set to 0:

$$f_i = \max\left\{0, F_i(\boldsymbol{x}_i^{\text{target}}) - D \cdot \left(\boldsymbol{x}_i^{\text{target}} - \boldsymbol{x}_i\right)^2\right\}. \tag{A.2}$$

The following subsections describe the three adaptation algorithms. For the fully automatic segmentation chain in Section 5.4, the adaptation is preceded by a localization step using the Generalized Hough Transform. This is described in the last subsection.

A.2. Global pose adaptation

We can treat the complete model as a rigid entity that can be rotated, translated, and also scaled. The degrees of freedom are then described by a *single similarity* transformation $\mathcal{T}_{\text{sim}}$, i.e., all vertex coordinates are transformed according to:

$$\boldsymbol{v}_k = \mathcal{T}_{\text{sim}}[\boldsymbol{m}_k]. \tag{A.3}$$

Here, $\boldsymbol{m}_k$ are vertex coordinates of a mean mesh. After boundary detection, the adapted vertex coordinates $\boldsymbol{v}_k$ are obtained by optimizing $\mathcal{T}_{\text{sim}}$ such that E_{ext} in (A.1) is minimized. (Note that the triangle centers $\boldsymbol{c}_i$ in (A.1) are averages of three vertices $\boldsymbol{v}_k$ and thus a function of $\mathcal{T}_{\text{sim}}$.) This optimization can be interpreted as a rigid registration with scaling. The procedure of boundary detection and registration is iterated until some convergence criterion is met.

A.3. Piecewise affine model adaptation

A much refined parametric adaptation can be obtained by treating P different model parts with independent affine transformations. For the task of cardiac image segmentation with a mesh comprising all chambers and other structures, this parametric representation was found to describe the major cardiac shape variations (Ecabert et al., 2006; Ecabert et al., 2008). It is of similar complexity as a point distribution model (PDM) (Cootes et al., 1994; Cootes et al., 1995) trained on a multitude of data sets. However, the model parts can be identified by an expert and the piecewise affine transformations can be assigned without any training images. Each part p (such as the left atrium or the right ventricle) is now free to undergo an affine transformation $\mathcal{T}_{\text{aff},p}$. In regions near transitions between different model parts, the affine transformations are interpolated to ensure connectivity and smooth transitions. Formally, we get:

$$\boldsymbol{v}_k = \sum_{p=1}^{P} w_{k,p} \cdot \mathcal{T}_{\text{aff},p}[\boldsymbol{m}_k]. \tag{A.4}$$

Here, $\sum_p w_{k,p} = 1$ for each vertex k. Now, the free parameters are those of all affine transformations $\mathcal{T}_{\text{aff},p}$. As for the pose adaptation with one global similarity transformation (A.3), we iteratively search target points and then optimize all transformation parameters with respect to the external energy (A.1).

A.4. Deformable model adaptation

To obtain a locally accurate segmentation, we adopt the shape-constrained deformable models introduced in Weese et al. (2001). This framework is based on a parametric shape model such as, e.g., (A.3) or (A.4). In addition to the parametric shape transformations, it allows local deformations. These are controlled by an internal energy which penalizes differences of edge lengths and orientations between the adapting mesh and the registered shape model. For the parametric shape model (A.4), we formulate the internal energy as follows:

$$E_{\text{int}} = \sum_{k=1}^{V} \sum_{\ell \in N(k)} \sum_{p=1}^{P} w_{k,p} \cdot \left((\boldsymbol{v}_k - \boldsymbol{v}_\ell) - \left(\mathcal{T}_{\text{aff},p}[\boldsymbol{m}_k] - \mathcal{T}_{\text{aff},p}[\boldsymbol{m}_\ell] \right) \right)^2. \tag{A.5}$$

Here, $N(k)$ is the set of edge-connected neighbor vertices for vertex k. Vertex coordinates $\boldsymbol{v}_k$ are now also treated as free parameters (in addition to the parameters of all $\mathscr{T}_{\mathrm{aff},p}$) and model adaptation minimizes a total energy:

$$E_{\mathrm{tot}} = E_{\mathrm{ext}} + \alpha \cdot E_{\mathrm{int}}. \tag{A.6}$$

Here, α balances the contribution of both energies. Again, boundary detection and energy minimization are iterated. In practice, the free parameters are updated in two steps. First, a piecewise registration of the mean shape $\{\boldsymbol{m}_k\}$ to the current (adapting) shape $\{\boldsymbol{v}_k\}$ is performed by minimizing (A.5) with respect to all $T_{\mathrm{aff},p}$. Then, the $T_{\mathrm{aff},p}$ are fixed and the vertex coordinates become free variables during the minimization of the total energy (A.6).

A.5. 3D Generalized Hough Transform

To automatically place the mesh in the image region containing the organ of interest, a coarse localization can be performed using the Generalized Hough Transform (GHT) (Ballard, 1981). To this end, a 3D version of the GHT has been developed and described in detail in Ecabert et al. (2008). It starts by converting the 3D image volume to an edge image that may be downsampled to a very coarse resolution for acceleration if demands on accuracy are limited (our implementation downsamples the images to voxels of 3 mm edge length). The algorithm operates by shifting a template of the model edges across the current edge image. Coinciding edge voxels between the current image and the shifted template with similar edge orientation are counted and the shift parameters with maximum count are taken as solution. For practical implementation, the detailed formalism from Ecabert et al. (2008, Section IV) uses so-called R-tables. In principle, further degrees of freedom such as rotation and scaling can be introduced into the matching algorithm. In 3D, exploring a full affine transformation space is prohibitive due the high-dimensional parameter space. Our implementation therefore estimates only translation (three parameters) and tests a small set of isotropic scaling operations. To account for shape variations in the matching process, the edge template is composed from edge models corresponding to a variety of individual organ shapes.

References

Ballard, D.H., 1981. Generalizing the Hough transform to detect arbitrary shapes. Pattern Recognition 13 (2), 111–122.

Bosch, H.G., Mitchell, S.C., Lelieveldt, B.P.F., Nijland, F., Kamp, O., Sonka, M., Reiber, J.H.C., 2001. Active appearance-motion models for endocardial contour detection in time sequences of echocardiograms. In: Sonka, M., Hanson, K.M. (Eds.), Proceedings of the SPIE Medical Imaging, vol. 4322, pp. 257–268.

Brejl, M., Sonka, M., 2000. Object localization and border detection criteria design in edge-based image segmentation: automated learning from examples. IEEE Transactions on Medical Imaging 19 (10), 973–985.

Chakraborty, A., Staib, L., Duncan, J.S., 1996. Deformable boundary finding in medical images by integrating gradient and region information. IEEE Transactions on Medical Imaging 15 (6), 859–870.

Cohen, L., Cohen, I., 1993. Finite-element methods for active contours models and balloons for 2-D and 3-D images. IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (11), 1131–1147.

Cootes, T.F., Taylor, C.J., 2001. Statistical models of appearance for medical image analysis and computer vision. In: Sonka, M., Hanson K.M. (Eds.), Proceedings of the SPIE Medical Imaging, vol. 4322, pp. 236–248.

Cootes, T.F., Hill, A., Taylor, C.J., Haslam, J., 1994. The use of active shape models for locating structures in medical images. Image and Vision Computing 12 (7), 355–366.

Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J., 1995. Active shape models – their training and application. Computer Vision and Image Understanding 61 (1), 38–59.

de Bruijne, M., Nielsen, M., 2004. Shape particle filtering for image segmentation. In: Barillot, C., Haynor, D.R., Hellier, P. (Eds.), Proceedings of the MICCAI, LNCS, vol. 3216. Springer, pp. 168–175.

de Bruijne, M., van Ginneken, B., Viergever, M.A., Niessen, W.J., 2003. Adapting active shape models for 3D segmentation of tubular structures in medical images. In: Information Processing in Medical Imaging. LNCS, vol. 2732. Springer, pp. 136–147.

Delingette, H., 1999. General object reconstruction based on simplex meshes. International Journal of Computer Vision 32, 111–142.

Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Classification, second ed. Wiley.

Ecabert, O., Peters, J., Weese, J., 2006. Modeling shape variability for full heart segmentation in cardiac CT images. In: Reinhardt, J.M., Pluim, J.P.W. (Eds.), Proceedings of the SPIE Medical Imaging, vol. 6144, pp. 61443R-1–61443R-12.

Ecabert, O., Peters, J., Schramm, H., Lorenz, C., von Berg, J., Walker, M.J., Vembar, M., Olszewski, M.E., Subramanyan, K., Lavi, G., Weese, J., 2008. Automatic model-based segmentation of the heart in CT images. IEEE Transactions on Medical Imaging 27 (9), 1189–1201.

Frangi, A.F., Niessen, W.J., Viergever, M.A., 2001. Three-dimensional modeling for functional analysis of cardiac images: a review. IEEE Transactions on Medical Imaging 20 (1), 2–25.

Giorgi, B., Dymarkowski, S., Maes, F., Kouwenhoven, M., Bogaert, J., 2002. Improved visualization of coronary arteries using a new three-dimensional submillimeter MR coronary angiography sequence with balanced gradients. American Journal of Roentgenology 179 (4), 901–910.

Heimann, T., Wolf, I., Meinzer, H.-P., 2006. Active shape models for a fully automated 3D segmentation of the liver – an evaluation on clinical data. In: Larsen, R., Nielsen, M., Sporring, J. (Eds.), Proceeding of MICCAI, LNCS, vol. 4191. Springer, pp. 41–48.

Heimann, T., Münzing, S., Meinzer, H.-P., Wolf, I., 2007. A shape-guided deformable model with evolutionary algorithm initialization for 3D soft tissue segmentation. In: Karssemeijer, N., Lelieveldt, B. (Eds.), Information Processing in Medical Imaging, LNCS, vol. 4584. Springer, pp. 1–12.

Heimann, T., 2007. Statistical Shape Models for 3D Medical Image Segmentation, Technical Report, Division of Medical and Biological Informatics, German Cancer Research Center, Heidelberg.

Kass, M., Witkin, A., Terzopoulos, D., 1988. Snakes: active contour models. International Journal of Computer Vision 1 (4), 321–331.

Kaus, M.R., Pekar, V., Lorenz, C., Truyen, R., Lobregt, S., Weese, J., 2003. Automated 3-D PDM construction from segmented images using deformable models. IEEE Transactions on Medical Imaging 22 (8), 1005–1013.

Kaus, M.R., von Berg, J., Weese, J., Niessen, W., Pekar, V., 2004. Automated segmentation of the left ventricle in cardiac MRI. Medical Image Analysis 8, 245–254.

Lorenz, C., von Berg, J., 2006. A comprehensive shape model of the heart. Medical Image Analysis 10 (4), 657–670.

Lötjönen, J., Kivistö, S., Koikkalainen, J., Smutek, D., Lauerma, K., 2004. Statistical shape model of atria, ventricles and epicardium short- and long-axis MR images. Medical Image Analysis 8, 371–386.

Manzke, R., Reddy, V.Y., Dalal, S., Hanekamp, A., Rasche, V., Chan, R.C., 2006. Intra-operative volume imaging of the left atrium and pulmonary veins with rotational X-ray angiography. In: Larsen, R., Nielsen, M., Sporring, J. (Eds.), Proceedings of the MICCAI, LNCS, vol. 4190. Springer, pp. 604–611.

Manzke, R., Meyer, C., Ecabert, O., Peters, J., Noordhoek, N.J., Thiagalingam, A., Reddy, V.Y., Chan, R.C., Weese, J., in press. Intra-procedural rotational X-ray imaging for atrial fibrillation ablation guidance, IEEE Transactions on Medical Imaging, doi:10.1109/TMI.2009.2021946. Available at: <http://ieeexplore.ieee.org/xpl/freepre_abs_all.jsp?arnumber=4967955>.

McInerney, T., Terzopoulos, D., 1996. Deformable models in medical image analysis. Medical Image Analysis 1 (2), 91–108.

Mitchell, S.C., Lelieveldt, B.P.F., van der Geest, R.J., 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 Transactions on Medical Imaging 20 (5), 415–423.

Montagnat, J., Sermesant, M., Delingette, H., Malandain, G., Ayache, N., 2003. Anisotropic filtering for model-based segmentation of 4D cylindrical echocardiographic images. Pattern Recognition Letters 24, 815–828.

Nyul, L.G., Udupa, J.K., 1999. On standardizing the MR image intensity scale. Magnetic Resonance in Medicine 42, 1072–1081.

Ozgun, M., Hoffmeier, A., Kouwenhoven, M., Botnar, R.M., Stuber, M., Scheld, H.H., Manning, W.J., Heindel, W., Maintz, D., 2005. Comparison of 3D segmented gradient-echo and steady-state free precession coronary mri sequences in patients with coronary artery disease. American Journal of Roentgenology 185 (1), 103–109.

Peters, J., Ecabert, O., Weese, J., 2005. Feature optimization via simulated search for model-based heart segmentation. In: Lemke, H.U., Inamura, K., Doi, K., Vannier, M.W., Farman, A.G. (Eds.), Proceedings of the CARS, ICS, vol. 1281. Elsevier, pp. 33–38.

Peters, J., Ecabert, O., Meyer, C., Schramm, H., Kneser, R., Groth, A., Weese, J., 2007. Automatic whole heart segmentation in static magnetic resonance image volumes. In: Ayache, N., Ourselin, S., Maeder, A. (Eds.), Proceedings of the MICCAI, LNCS, vol. 4792. Springer, pp. 402–410.

Peters, J., Ecabert, O., Lorenz, C., von Berg, J., Walker, M.J., Ivanc, T.B., Vembar, M., Olszewski, M.E., Weese, J., 2008. Segmentation of the heart and major vascular structures in cardiovascular CT images. In: Pluim, J.P.W., Reinhardt, J.M. (Eds.), Proceedings of the SPIE Medical Imaging, vol. 6914, pp. 691417-1–691417-12.

Sakalli, M., Lam, K.-M., Yan, H., 2006. A faster converging snake algorithm to locate object boundaries. IEEE Transactions on Image Processing 15 (5), 1182–1191.

Sermesant, M., Forest, C., Pennec, X., Delingette, H., Ayache, N., 2003. Deformable biomechanical models: application to 4D cardiac image analysis. Medical Image Analysis 7, 475–488.

Sermesant, M., Moireau, P., Camara, O., Sainte-Marie, J., Andriantsimiavona, R., Cimrman, R., Hill, D.L.G., Chapelle, D., Razavi, R., 2006. Cardiac function estimation from MRI using a heart model and data assimilation: advances and difficulties. Medical Image Analysis 10 (4), 642–656.

Sonka, M., Fitzpatrick, J.M. (Eds.), 2000. Handbook of Medical Imaging. Medical Image Processing and Analysis, vol. 2. SPIE Press. Chapter 3.

Staib, L.H., Duncan, J.S., 1992. Boundary finding with parametically deformable models. IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (11), 1061–1075.

Sukno, F.M., Frangi, A.F., 2008. Reliability estimation for statistical shape models. IEEE Transactions on Image Processing 17 (12), 2442–2455.

Sukno, F.M., Ordás, S., Butakoff, C., Cruz, S., Frangi, A.F., 2007. Active shape models with invariant optimal features: application to facial analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (7), 1105–1117.

Tobon-Gomez, C., Butakoff, C., Aguade, S., Sukno, F., Moragas, G., Frangi, A.F., 2008. Automatic construction of 3D-ASM intensity models by simulating image acquisition: application to myocardial gated SPECT studies. IEEE Transactions on Medical Imaging 27 (11), 1655–1667.

van Assen, H.C., Danilouchkine, M.G., Frangi, A.F., Ordás, S., Westenberg, J.J.M., Reiber, J.H.C., Lelieveldt, B.P.F., 2006. SPASM: a 3D-ASM for segmentation of sparse and arbitrarily oriented cardiac MRI data. Medical Image Analysis 10, 286–303.

van Assen, H.C., Danilouchkine, M.G., Dirksen, M.S., Reiber, J.H.C., Lelieveldt, B.P.F., 2008. A 3-D active shape model driven by fuzzy inference: application to cardiac CT and MR. IEEE Transactions on Information Technologt and Biomedicine 12 (5), 595–605.

van Ginneken, B., Frangi, A.F., Staal, J.J., ter Haar Romeny, B.M., Viergever, M.A., 2002. Active shape model segmentation with optimal features. IEEE Transactions on Medical Imaging 21 (8), 924–933.

von Berg, J., Lorenz, C., 2005. Multi-surface cardiac modelling, segmentation and tracking. In: Frangi, A.F., Radeva, P., Santos, A., Hernandez, M. (Eds.), Proceedings of the Functional Imaging and Modeling of the Heart, LNCS, vol. 3504. Springer, pp. 1–11.

Weese, J., Kaus, M.R., Lorenz, C., Lobregt, S., Truyen, R., Pekar, V., 2001. Shape constrained deformable models for 3D medical image segmentation. In: Insana, M.F., Leahy, R.M. (Eds.), Information Processing in Medical Imaging, LNCS, vol. 2082. Springer, pp. 380–387.

Wierzbicki, M., Moore, J., Drangova, M., Peters, T., 2008. Subject-specific models for image-guided cardiac surgery. Physics in Medicine and Biology 53, 5295–5312.

Zhang, J., Zhou, S.K., Comaniciu, D., McMillan, L., 2008. Discriminative learning for deformable shape segmentation: a comparative study. In: Forsyth, D., Torr, P., Zisserman, A. (Eds.), Proceedings of the ECCV, LNCS, vol. 5302. Springer, pp. 711–724.

Zheng, Y., Barbu, A., Georgescu, B., Scheuering, M., Comaniciu, D., 2008. Four-chamber heart modeling and automatic segmentation for 3-D cardiac CT volumes using marginal space learning and steerable features. IEEE Transactions on Medical Imaging 27 (11), 1668–1681.

Zhuang, X., Rhode, K., Arridge, S., Razavi, R., Hill, D., Hawkes, D., Ourselin, S., 2008. An atlas-based segmentation propagation framework using locally affine registration – application to automatic whole heart segmentation. LNCS, vol. 5242. Springer.