Contents lists available at ScienceDirect





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



Real-time 3D interactive segmentation of echocardiographic data through user-based deformation of B-spline explicit active surfaces

Daniel Barbosa<sup>a,b,c,\*</sup>, Brecht Heyde<sup>a</sup>, Maja Cikes<sup>a</sup>, Thomas Dietenbeck<sup>b</sup>, Piet Claus<sup>a</sup>, Denis Friboulet<sup>b</sup>, Olivier Bernard<sup>b</sup>, Jan D'hooge<sup>a</sup>

<sup>a</sup> Laboratory on Cardiovascular Imaging and Dynamics, Department of Cardiovascular Sciences, KU Leuven, Belgium

<sup>b</sup> Université de Lyon, CREATIS, CNRS UMR5220, Inserm U1044, INSA-Lyon, Université Lyon 1, France

<sup>c</sup> DIGARC, Polytechnic Institute of Cávado and Ave, Barcelos, Portugal

## ARTICLE INFO

Article history: Received 7 January 2013 Received in revised form 27 August 2013 Accepted 8 October 2013

Keywords: Active contours B-spline explicit active surfaces User interaction Interactive segmentation

## ABSTRACT

Image segmentation is an ubiquitous task in medical image analysis, which is required to estimate morphological or functional properties of given anatomical targets. While automatic processing is highly desirable, image segmentation remains to date a supervised process in daily clinical practice. Indeed, challenging data often requires user interaction to capture the required level of anatomical detail. To optimize the analysis of 3D images, the user should be able to efficiently interact with the result of any segmentation algorithm to correct any possible disagreement. Building on a previously developed realtime 3D segmentation algorithm, we propose in the present work an extension towards an interactive application where user information can be used online to steer the segmentation result. This enables a synergistic collaboration between the operator and the underlying segmentation algorithm, thus contributing to higher segmentation accuracy, while keeping total analysis time competitive. To this end, we formalize the user interaction paradigm using a geometrical approach, where the user input is mapped to a non-cartesian space while this information is used to drive the boundary towards the position provided by the user. Additionally, we propose a shape regularization term which improves the interaction with the segmented surface, thereby making the interactive segmentation process less cumbersome. The resulting algorithm offers competitive performance both in terms of segmentation accuracy, as well as in terms of total analysis time. This contributes to a more efficient use of the existing segmentation tools in daily clinical practice. Furthermore, it compares favorably to state-of-the-art interactive segmentation software based on a 3D livewire-based algorithm.

© 2013 Elsevier Ltd. All rights reserved.

# 1. Introduction

Image segmentation is an ubiquitous task in medical imaging analysis. Indeed, the quantification of morphological or functional properties of a given organ or tissue always involves its identification in the image domain, either by identifying the boundaries of anatomical structures or by defining a region for subsequent analysis. Therefore, the role of image segmentation is central in any medical image analysis workflow. This task has been initially addressed through manual delineation by a user of the desired object in the image. However, with the ever increasing amount of 3D imaging modalities, a simple slice-by-slice 2D manual processing of medical image data becomes a very time consuming,

\* Corresponding author at: Laboratory on Cardiovascular Imaging and Dynamics, Department of Cardiovascular Sciences, KU Leuven, Belgium. Tel.: +35 1961864665. *E-mail addresses*: daniel.barbosa@uzleuven.be,

djcbarbosa@gmail.com (D. Barbosa).

tedious and clinically inefficient solution for routine practice. Furthermore, manual delineation of anatomical structures in 3D data is prone to significant intra and inter-observer variability. It should also be noted that 3D contouring through slice-by-slice 2D processing is not trivial, since boundary continuity in the out-of-plane direction is not explicitly enforced. Therefore, computational tools are crucial to aid the clinicians in this task and there is a vast literature on diverse methods suited to alleviate the need of fully manual analysis. Naturally, fully automatic image segmentation is highly desirable, and several powerful concepts such as active shape models, statistical atlases and other knowledge-based modeling have been proposed to automatically segment organs within 3D volumetric data. Nonetheless, due to inter-subject variability in anatomy and due to changes in imaging conditions, such automatic approaches are very challenging, and often need manual correction. Furthermore, medical image segmentation remains a supervised process in clinical practice and thus the segmentation result should reflect the clinicians' vision of the anatomical structures. Thus, the user should be able to quickly and efficiently interact with the result

<sup>0895-6111/\$ -</sup> see front matter © 2013 Elsevier Ltd. All rights reserved. http://dx.doi.org/10.1016/j.compmedimag.2013.10.002

of any segmentation algorithm to correct any possible disagreement.

User interaction in medical image segmentation has been reviewed by Olabarriaga and Smeulders [1]. The user input can be introduced at different levels of the processing pipeline, either by initializing and setting the parameters of the segmentation algorithm or by interacting with it, allowing online steering of the segmentation result. The segmentation algorithms that follow the first approach can be broadly classified as semi-automatic, where the segmentation process has a 'fire and forget' behavior [2]. On the other hand, the second approach more closely resembles a truly interactive process, where the algorithm reacts to the information introduced by the user preferably in real-time. While the first corresponds to a more automatized approach, with consequently less inter-observer variability, the control of the segmentation outcome by contour initialization and parameter setting might not be sufficient in more challenging areas of the image, where direct user input is required [3]. This can lead to a cumbersome tuning process, which requires user's knowledge of the underlying algorithm. On the other hand, purely interactive segmentation methods may require too much user input, which could affect the intra- and interobserver variability, and require near real-time processing of the segmentation method for optimal user steering. An efficient image segmentation algorithm is characterized by having a computational part that is fast, accurate, highly autonomous and predictable and whose user interventions are few, quick and simple [1]. Therefore, a hybrid approach, using initial (semi-) automatic segmentation with subsequent interactive real-time correction could provide an interesting trade-off for optimal clinical application.

Several algorithms are able to include user clues in the segmentation process. The most simple approaches are geometrical modeling tools which can be used to aid the manual construction of a 3D model, such as in [4,5]. More recently, Heckel et al. have proposed a variational interpolation framework to implement near real-time reconstruction of a 3D model from a set of contours [2]. On cardiac segmentation, the ellipsoidal shape of the LV can be used to improve the efficiency of the generation of the 3D model, as in the guide-point modeling algorithm from Young et al. [6]. Obviously, these methods are relying solely on heavy user input, therefore not directly exploiting image information.

The classical example of the interactive segmentation driven by the user paradigm is the livewire algorithm and its variants. In this method, the user sets a number of sequential seed points, while the optimal cost path between them is estimated via graph-based analysis, such as Dijkstra's algorithm [7]. Several extensions have been proposed for 3D segmentation [8–10] and algorithms for efficient 3D livewire interactive segmentation of complex topology objects also exist [11]. However, the extension of the livewire framework to 3D is not straightforward, as the interaction paradigm is intrinsically 2D, since the optimal path remains a 1D entity. Although the recent work of Grady on minimal weight surfaces provides a truly three-dimensional extension of shortest path-based methods [12], 3D interactive segmentation using livewire-based approaches still rely on a collection of 2D contours from where a 3D model needs to be constructed.

Graph-Cuts, originally proposed by Boykov et al. [13,14], and its variants offer an alternative paradigm for interactive segmentation. In this framework, the user introduces seed points that will correspond to different regions (object/background) and the segmentation result is calculated as the optimal partition graph between the object and the background, which can be efficiently computed using the max-flow/min-cut algorithm [15]. One important advantage offered by this framework is its computational efficiency, as well as the guarantee that the segmentation result corresponds to the global energy minimum of the segmentation criterion. By interactively placing additional seeds, the user can efficiently interact with the segmentation results. However, these seed points correspond to hard constraints on the segmented regions and not to boundary positions, which can make precise boundary manipulation cumbersome to the user.

Deformable models are a very popular family of segmentation methods, arising from the seminal work of Kass et al. [16]. In the original formulation of snakes, user input points could be used as attractor/repulsor sinks for the parametric curve evolution. ITK-SNAP is a particularly popular software tool that offers several options for 3D active contour segmentation, where the user can control the parameters of the segmentation algorithm and visually follow the evolution of the segmentation process, enabling online re-tuning of the parameters [17]. Therefore, ITK-SNAP is a perfect example for the type-I interaction as defined by Olabarriaga and Smeulders. More recently, Delgado-Gonzalo et al. have proposed a 3D parametric snake where user interaction plays an important role, since the compact parametrization of the surface allows shape deformation through the manipulation of a limited number of control points [18]. Interestingly, in the work of Liang et al. a unifying framework between snakes and livewire is proposed, allowing to take the complimentary advantages of these up until then competing techniques [19]. It should be noted that interactive free-form surface editing operators are also available for level-set based segmentation algorithms [20]. However, 3D segmentation using level-set approaches is computationally intensive, often requiring advanced GPU-based implementations to achieve acceptable running times for near real-time applications [21,22].

We propose in this paper to take advantage of the existing framework for real-time segmentation of challenging inhomogeneous 3D data, introduced very recently by the authors [23], in order to design a hybrid process smoothly integrating automatic segmentation and real-time interaction. The segmentation method uses B-spline explicit active surfaces (BEAS) to recover objects from volumetric data, allowing to use both global or local region-based segmentation energies to evolve the contour. In spite of obtaining promising results, the initial semi-automated algorithm relied on manual input (6 clicks per 3D volume) to fit an ellipsoid used to initialize the segmentation algorithm [24]. Since manual initialization accounted for the vast majority (~95%) of the total analysis time, a fast and automatic initialization method was posteriorly developed, allowing fully automatic segmentation [25].

Although the existing BEAS framework offers very competitive performance, the supervised nature of the segmentation process introduces the need of pursuing strategies to efficiently allow the user to correct the segmentation results if needed. The manual correction of the segmentation results provided by BEAS can be efficiently steered by the user by evolving the segmented surface towards the additional points introduced by the operating physician. This is mathematically straightforward thanks to BEAS explicit parametric formulation. Furthermore, the computational speed of the method allows a truly interactive experience, similar to the livewire segmentation paradigm, with the advantage of explicitly deforming the surface rather than a 1D path.

The originality of the proposed method is twofold: first, a new energy term is introduced in the variational formulation of the method, in order to include the contour information introduced by the user. The proposed energy term can be easily extended to the case of multiple points being annotated by the operating physician. Secondly, we introduce a regularization term which yields a geometrically optimal interaction with the surface. This allows to propagate the input information in a geometrically smooth manner and to avoid a too localized deformation of the surface in the neighborhood of the points introduced by the user. Thanks to the computational efficiency of the underlying BEAS segmentation algorithm, the resulting tool offers a synergistic solution between a fully automatic segmentation tool and a real-time interactive segmentation method, where the user can introduce additional points to steer the segmentation process. In this paper we focus on the application of the proposed framework to real-time 3D echocardiographic (RT3DE) data. Note however that this framework is generic and can thus be easily used for a wide class of 3D segmentation tasks.

The paper is structured as follows. In Section 2, we focus on the general formulation of image segmentation problems using BEAS and we introduce the proposed variational formulation user interaction term. Also the geometric regularizer enabling a more efficient surface manipulation is presented here. In Section 3, we discuss some implementation issues of our method. In Section 4, we evaluate the performance of the method using RT3DE data. Furthermore, a benchmark comparison against a state-of-the-art 3D interactive segmentation software is equally presented. The key findings are discussed in depth in Section 5, while the study limitations are highlighted in Section 6. Lastly, we give the main conclusions and perspectives of this work in Section 7.

# 2. Methods

# 2.1. B-spline explicit active surfaces

The segmentation framework (BEAS) used in the present work has been recently proposed by the authors in order to allow realtime segmentation of challenging inhomogeneous 3D data [23]. The fundamental concept of this method is to regard the boundary of an object as an explicit function in a generic coordinate system, where one of the coordinates of the points within the surface is given explicitly as a function of the remaining coordinates [26]. Such explicit relation can be mathematically defined as:

$$\psi: \mathbb{R}^{n-1} \mapsto \mathbb{R}, (x_2, \dots, x_n) \mapsto x_1 = \psi(\mathbf{x}^*), \tag{1}$$

where  $\mathbf{x} \in \mathbb{R}^n$  is a point of coordinates  $\{x_1, \ldots, x_n\}$  in an n-dimensional space and  $\mathbf{x}^* \in \mathbb{R}^{n-1}$  is a point of coordinates  $\{x_2, \ldots, x_n\}$  in the associated (n-1)-dimensional subspace.

In this framework, the explicit function  $\psi$  is expressed as the linear combination of B-spline basis functions [27]:

$$x_1 = \psi(x_2, \dots, x_n) = \sum_{\mathbf{k} \in \mathbb{Z}^{n-1}} c[\mathbf{k}] \beta^d \left(\frac{\mathbf{x}^*}{h} - \mathbf{k}\right),$$
(2)

where  $\beta^d(\cdot)$  is the uniform symmetric n-1-dimensional B-spline of degree d. This function is separable and is built as the product of n-1 one-dimensional B-splines, so that  $\beta^d(\mathbf{x}^*) = \prod_{j=2}^n \beta^d(x_j)$ . The knots of the B-splines are located on a rectangular grid defined on the chosen coordinate system, with a regular spacing given by h. The coefficients of the B-spline representation are gathered in  $c[\mathbf{k}]$ . Note that this representation follows what has been initially proposed for the level-set framework by Bernard et al. [28]. This B-spline formalism offers several important methodological advantages, since the contour evolution can be expressed as a separable convolution, which translates in a faster and more robust segmentation result than using other basis functions, such as radial basis functions (RBF) [29].

As shown in [23], both global and local region-based segmentation energy terms can be used in the BEAS framework. Although generic data attachment terms can be used, such as the Chan–Vese [30] and Yezzi [31] segmentation functionals and their localized counterparts [32], the segmentation energy can be tailored for more specific applications. For instance, the darker appearance of the blood in RT3DE data can be explicitly taken into account when designing the energy to be minimized during the segmentation of RT3DE data, which improves the robustness of the method in this particular application [25]. Taking this an example, it can be



Fig. 1. Left ventricular shape parametrization in the spherical domain.

shown that this energy can be directly minimized wrt. the B-spline coefficients, using the following evolution equations:

$$c[\mathbf{k}]^{(t+1)} = c[\mathbf{k}]^{(t)} - \lambda \frac{\partial E_L}{\partial c[\mathbf{k}]^{(t)}},$$
(3)

$$\frac{\partial E_L}{\partial c[\mathbf{k}_1]} = \int_{\Gamma} \left( \frac{\left( \bar{l}(\mathbf{x}^*) - u_{\mathbf{x}} \right)}{A_u} + \right)$$
(4)

$$\frac{\left(\bar{I}(\mathbf{x}^*)-v_{\mathbf{x}}\right)}{A_{\nu}})\beta^d\left(\frac{\mathbf{x}^*}{h}-\mathbf{k}_i\right)\,\mathrm{d}\mathbf{x}^*,$$

where  $A_u$  and  $A_v$  are the areas inside and outside the interface  $\Gamma$  used to estimate the local means  $u_x$  and  $v_x$ , respectively. For clarity sake,  $\overline{I}(x^*)$  corresponds to the image value at the position  $\mathbf{x} = \{\psi(\mathbf{x}^*), x_2, ..., x_n\}$ . In the present work, the  $\psi$  function was defined in spherical coordinates and the LV parametrization can be seen in Fig. 1. The segmentation algorithm is automatically initialized with an ellipsoid using the algorithm proposed in [25], allowing fully automatic segmentation of the left ventricle. Further details on the foundations of the BEAS framework can be found in [23].

## 2.2. Surface deformation via user-based online steering

Since image segmentation remains a supervised process in clinical practice, the operating physician must be able to interact with the segmentation tools to ensure that its results capture all the desired anatomical details. To this end we propose to incorporate user clues to adapt the segmentation results to the positions where, in the physician's point of view, the contour should pass. These points will act as anchors attracting the surface. By interactively introducing additional user points to the segmentation result and by dragging them towards the proper position, the physician can efficiently interact with the segmentation result and thanks to the computational efficiency of BEAS, there will be real-time feedback of the effect of these modifications. This allows an online interaction, which resembles the behavior of a 3D livewire algorithm, where in this case the user interacts directly with the entire surface rather than with a 2D contour.

Recalling that our generic explicit function  $\psi(\mathbf{x}^*)$  corresponds to a surface whose radius is a function of both azimuthal and zenithal angles (*i.e.*  $\rho = \psi(\theta, \varphi)$ ), a point introduced by the user can equally be expressed in spherical coordinates (*i.e.*  $p_{user} = (\rho_{user}, \theta_{user}, \varphi_{user})$ ).



Fig. 2. Impact of geometrical regularization on the user-based deformation of the 3D segmented surface ((a) fully automatic segmentation result; (b) and (c) direct user manipulation; (d) geometrically regularized user interaction).

By simply penalizing the parametric distance between the current surface position and the point position desired by the user (*i.e.*  $D = (\rho - \rho_{user})^2$ ), we can easily evolve the surface towards the user's desired position. In this case, the energy term driving the surface can expressed as:

$$E_{user} = \int \int_{\Gamma} \delta\left((\theta, \varphi) - (\theta_{user}, \varphi_{user})\right) \left(\psi(\theta, \varphi) - \rho_{user}\right)^2 \mathrm{d}\theta \,\mathrm{d}\varphi, \quad (5)$$

where  $(\rho_{user}, \theta_{user}, \varphi_{user})$  corresponds to the point introduced by the user expressed in spherical coordinates.  $\delta((\theta, \varphi) - (\theta_{user}, \varphi_{user}))$ corresponds to the 2D Dirac delta function arising from product  $\delta(\theta - \theta_{user})\delta(\varphi - \varphi_{user})$ , which is non-zero only at the position  $\mathbf{x}^* = (\theta_{user}, \varphi_{user})$ . As shown in the appendix,  $E_{user}$  can be directly minimized wrt. to each B-spline coefficient  $c[\mathbf{k}_i]$ , using the following evolution equation:

$$\frac{\partial E_{user}}{\partial c[\mathbf{k}_i]} = 2 \int \int_{\Gamma} \delta\left((\theta, \varphi) - (\theta_{user}, \varphi_{user})\right) (\psi(\theta, \varphi) - \rho_{user}) \\ \times \beta^d \left(\frac{(\theta, \varphi)}{h} - \mathbf{k}_i\right) d\theta d\varphi.$$
(6)

In the case of the user inputing several points, the aforementioned energy can be also employed. To this end, we will take the sum of the parametric distance between the current surface and the points indicated by the user. Then, the segmented surface can be deformed towards multiple user points using the following evolution equation for each B-spline coefficient  $c[\mathbf{k}_i]$ ,:

$$\frac{\partial E_{user}}{\partial c[\mathbf{k}_i]} = 2 \sum_{k=1}^{N_p} \int \int_{\Gamma} \delta\left((\theta, \varphi) - (\theta_k, \varphi_k)\right) (\psi(\theta, \varphi) - \rho_k) \\ \times \beta^d \left(\frac{(\theta, \varphi)}{h} - \mathbf{k}_i\right) d\theta d\varphi.$$
(7)

where  $N_p$  stands for the number of points introduced by the user.

# 2.3. Surface geometrical regularization for optimal user interaction

The direct application of the proposed user-based online steering framework is straightforward but results in a "bump-like" modification of the segmented surface, as shown in Fig. 2(b). This arises from the fact that the energy proposed to drive the surface towards the point input by the user has a local nature. In fact, the energy minimization process drives the surface only along the radial direction for the azimuthal and zenithal angles of the point introduced by the user. On the other hand this point-wise modification of the surface affects the B-spline coefficients locally, as shown in the energy gradient wrt. to each B-spline coefficient expressed in Eq. (7), which is different than zero only in a neighborhood around the introduced point. This local neighborhood will depend on the scale of the B-spline function supporting the surface and also on the sampling grid used to discretize the problem.

Although this behavior could be tackled by introducing more user points, it would increase the time needed to efficiently correct the segmented surface (Fig. 2(c)). In order to reduce the



Fig. 3. Oversmoothing of anatomical details at the LV base due to excessive geometrical regularization ((a) original LV surface; (b) direct user manipulation; and (c) geometrically regularized user interaction).

resulting bumps, we propose to regularize the surface by extending the influence of a given input point beyond the local B-spline kernel support. To this end, we propose to add a surface diffusion flow term to the interface evolution equation, which will evolve the surface towards the minimization of curvature variation in the surface. We follow the formulation originally proposed by Delingette and Montagnat for 2D parametric contours [33]. As detailed in the Appendix, expanding this approach to the 3D case implies using a flow  $F_R$  defined as:

$$F_R(\mathbf{x}^*) = -\Delta \kappa_M(\mathbf{x}^*),\tag{8}$$

where  $\kappa_M(\mathbf{x}^*)$  is the mean curvature of the interface  $\Gamma$  at the position  $\mathbf{x}^*$ . The typical notation for the surface mean curvature (*i.e.* H) was not used to avoid confusion with the Heaviside operator. As noted in [33], the evaluation of  $\Delta \kappa_M$  may be computed in practice using the following approximation:

$$\Delta \kappa_M(\mathbf{x}^*) = \frac{1}{\Omega_A} \int_{\Omega_A} \kappa_M(\mathbf{x}^*) dA - \kappa_M(\mathbf{x}^*), \tag{9}$$

where  $\Omega_A$  represents the local area where  $\kappa_M(\mathbf{x}^*)$  is averaged. This geometrical regularization term can be interpreted as a force term driving the surface towards the local average of its mean curvature, thus penalizing local curvature variation.

As shown in the Appendix, the previous geometrical regularization term can be used to evolve the BEAS surface using the following equation:

$$\frac{\partial E_R}{\partial c[\mathbf{k}_i]} = \int_{\Gamma} ||\nabla \phi(\mathbf{x}^*)|| \Delta \kappa_M(\mathbf{x}^*) \beta^d \left(\frac{\mathbf{x}^*}{h} - \mathbf{k}_i\right) d\mathbf{x}^*.$$
(10)

By adding the proposed geometrical regularization term to the user-based energy introduced in the previous section, the information introduced by the user will be propagated to regions outside the local B-spline kernel support, while keeping a smooth surface with minimal curvature variation (Fig. 2(d)). We should note however, that (10) will evolve the segmented surface towards a sphere (*i.e.* a surface with minimal curvature variation), which might negatively affect the automatic segmentation results, as shown in Fig. 3. A solution to overcome this problem is presented in Section 3.

## 2.4. Interactive 3D real-time segmentation

Starting from the fully automatic segmentation results provided by the algorithm presented in [25], the user is allowed to introduce points in order to manually steer the segmentation where the automatic algorithm failed to capture the correct position of the LV boundary. The update of the segmented surface was achieved through the direct minimization of the energy terms with respect to the B-spline coefficients  $c[\mathbf{k}]$ , yielding:

$$c[\mathbf{k}]^{(\tau+1)} = c[\mathbf{k}]^{(\tau)} - \frac{\partial E}{\partial c[\mathbf{k}]^{(\tau)}}$$
(11)

$$\frac{\partial E}{\partial c[\mathbf{k}_i]} = \alpha_1 \frac{\partial E_{user}}{\partial c[\mathbf{k}_i]} + \alpha_2 \frac{\partial E_R}{\partial c[\mathbf{k}_i]},\tag{12}$$

where  $\alpha_1$  and  $\alpha_2$  are the hyperparameters controlling the relative weight of the user-based and geometrically-based energy terms in the surface segmentation result. Note that in the present interactive segmentation algorithm no data attachment term is included. However, image-based forces could also be added to the interface evolution.

# 3. Implementation details

#### 3.1. Segmentation parameters

The following settings were applied for all experiments:

- We used a cubic B-spline function as basis for the BEAS representation. This function provides a good trade-off between smoothing properties and computational cost. The scale parameter was set to *h* = 2; The size of the neighborhood used to estimate the local means was set to 16.
- As previously mentioned, the ψ function was defined in spherical coordinates and the LV surface was discretized in 24 × 16 points along the zenithal and azimuthal directions respectively.
- For the current work, the B-spline coefficients,  $c(\mathbf{k})$ , are gathered in a 2D index array, spanning the spherical domain with the scale step previously defined, resulting in  $(N_{\phi} \times N_{\theta})/h = (24 \times 16)/2$  B-spline coefficients.
- The parameters that adjust the relative strength between the information introduced by the user and the geometrical regularization term were set to  $\alpha_1 = 1$  and  $\alpha_2 = 0.25$ ;

These parameters were set empirically towards an optimal response of the interactive segmentation algorithm.

#### 3.2. ROI definition for geometrical regularization

The direct application of the proposed geometrical regularization scheme may lead to the loss of some anatomical details present in the automatic segmentation result. To this end, and to avoid oversmoothing in regions far away from the points introduced by the user, the proposed regularization scheme is only applied in a ROI around each point introduced by the user. A particularly sensitive



**Fig. 4.** Region of interest definition for improved geometrical regularization of the user interaction.

region is the basal plane, whose surface boundaries correspond to regions of high curvature variation. To this end, the proposed geometrical regularization is not applied in this surface area, as illustrated in Fig. 4. The base plane is defined as the cut along  $\varphi$ which has the larger average surface mean curvature. Naturally, this is an application specific adaptation, which demonstrate the flexibility of the underlying interactive segmentation method to be taylored to the application needs.

One should note that defining directly the ROI in the spherical domain (*i.e.*  $N \times M$  radians along the azimuthal and zenithal angles respectively) will lead to spatially varying ROI whose area will depend on the position of the point introduced by the user. This fact stems from the mismatch between the roughly ellipsoid shape of the left ventricle and the spherical sampling of the segmented surface, as shown Fig. 5(a) and (b). This is particularly evident along the zenithal (*i.e.* longitudinal) direction. To compensate this, we propose to define the geometrical regularization ROI based on



Fig. 5. Constant angle versus constant area formulation of the geometrical regularization region of interest (top: constant angle; bottom: constant area)

the local surface area, which is analytically computed during the automatic segmentation and thus available free of additional computational cost. It should be noted that although the surface is defined continuously, the local area values are defined over the discretization grid in the spherical domain used to computationally implement BEAS.

In order to simplify the local area estimation for the geometrical regularization ROI definition, we take a fixed size along the azimuthal (*i.e.* circumferential) direction corresponding to  $\pi/4$  and we dynamically adapt the size of the ROI along the zenithal (*i.e.* longitudinal) direction. The size of the ROI along the zenithal direction is such that the area of the ROI remains nearly constant in the discretized grid as highlighted in Fig. 5(c) and (d). The zenithal length of the area patch was kept to 1/3 of the average longitudinal arclength of the LV surface. Nonetheless it is important to stress the intrinsic quantization effect caused by the discrete implementation of BEAS.

#### 4. Experiments

# 4.1. Data acquisition and analysis

RT3DE data was acquired using a GE E9 scanner (GE Vingmed, Horten, Norway) equipped with a 4V transducer. The data used in the present study has been acquired at the enrollment of patients at UZ Leuven in a large ongoing multi-center clinical study (DOPPLER-CIP). The patients enrolled in this clinical study have a clinical profile corresponding to suspected chronic ischemic heart disease. The data used in the present work was taken randomly from the DOPPLER-CIP study database where the only inclusion criteria was to include patients which had been scanned at the UZ Leuven hospital for RT3DE acquisitions. This study was performed according to the ethical principles for medical research involving human subjects of the World Medical Association's declaration of Helsinki.

Each sequence was analyzed by two clinical experts, which have provided manual contours as well as interactive segmentations (using the proposed method and a state-of-the-art interactive segmentation tool) of the LV at the end-diastolic frame. Image quality was equally assessed by the two experts as poor (1), fair (2) or good (3) in accordance to the percentage of the myocardial wall clearly visible in the image (<60%, 60–75%, >75%), the contrast between the blood pool and the tissue and the presence of severe image artifacts.

Manual contouring was done using an in-house developed software package (Speqle3D, KU Leuven) dedicated to 3D visualization and manual delineation of LV boundaries [34]. This software package is implemented in MATLAB (The Mathworks Inc., Natick, MA). The segmented LV surfaces at end-diastole were used as reference for the experiments in the present paper. The mean of the enddiastolic volumes (EDV) estimated by the two experts was taken as reference value. Also the mean contour, defined as the isosurface comprising the local maxima of the distance function between both expert contours, was used to further assess the segmentation accuracy.

In order to test the performance of the interactive segmentation framework introduced in the present paper, the proposed algorithm (interactiveBEAS) was added to the existing software suite. By comparing the segmentation results of the proposed interactive method with the underlying automatic segmentation algorithm, the added value of the proposed interactive framework can be quantified. In order to benchmark the performance of the proposed framework, the two clinical experts were also asked to segment the end-diastolic frame of each RT3DE exam using TurtleSeg (www.TurtleSeg.org), a freely available tool implementing a 3D livewire-based interactive segmentation algorithm [35,36,11]. The so-obtained 2 different interactive segmentation solutions (interactiveBEAS and TurtleSeg) were compared with the

#### Table 1

Image quality evaluation on the analyzed dataset (#=27).

Image quality Average score	Number of exams	Percentage of the analyzed dataset
1 – Poor	6	22.22%
1.5 – Poor/fair	4	14.82%
2 – Fair	9	33.33%
2.5 – Fair/good	2	7.69%
3 – Good	6	22.22%

manually extracted references for accuracy assessment, in terms of LV volume accuracy, LV segmentation accuracy and analysis time.

#### 4.2. Results

Twenty-seven exams were used in the present study [EDV range: 72.6–164.3 ml]. The average image quality scores from the clinical experts are shown in Table 1. Note that a score of 1.5 would represent a case where one expert classified the image quality as poor (1), while the other expert classified it as fair (2). 22.22% (6/27) of the analyzed data was consensually classified as having poor image quality, while the same number of exams was considered to have good image quality. The total analysis times using the different software tools are shown in Table 2. To illustrate the segmentation improvement offered by the proposed interactive method, we show in Fig. 6 the increased agreement between the interactively segmented surfaces and the reference surface for the two poorest automatic segmentation results according to the Dice coefficient.

The performance of the underlying automatic segmentation algorithm and both interactive segmentation methods for the assessment of EDV is shown in Table 3. The proposed interactive segmentation algorithm shows stronger correlation with the reference EDV values. For both interactive approaches, a statistically significant bias was found for one of the users. The proposed interactive approach also presented tighter limits of agreement, although being statistically significant thighter than TurtleSeg only for one of the users (p < 0.05, f-test).

The surface segmentation error for the different methods is given in Table 4. A consistent tendency to improved segmentation performance was found using the proposed interactive segmentation method, although this improvement was only statistically significant for one of the users (p < 0.05, paired *t*-test).

The results of the inter-observer variability analysis for the EDV estimation using the different segmentation approaches can be found in Table 5. Although not statistically significant, the proposed interactive segmentation approach presented a tendency to reduce the bias between both users. Furthermore, the variability of the absolute EDV difference between the experts was statistically significantly reduced using interactiveBEAS when compared to manual contouring (p<0.05, f-test).

The inter-observer variability analysis for the different surface segmentation error metrics can be found in Table 6. Interactive-BEAS significantly reduces the inter-observer variability between the experts since statistically significant differences were found between interactiveBEAS and both manual contouring and segmentation using TurtleSeg for all the metrics (p < 0.05, paired *t*-test). This observation did not hold for TurtleSeg, despite a

moderate reduction of the inter-observer HD when compared to manual analysis.

#### 5. Discussion

Despite the efforts towards fully automatic segmentation in the medical image processing community, at current image segmentation remains a supervised process in clinical routine. It therefore remains crucial to allow user input in any segmentation tool to be clinically used in order to modify incorrect segmentation in an easy manner. This was the main motivation behind the present work in which an existing segmentation framework was modified in order to allow convenient user interaction.

By comparing the segmentation performance of the proposed interactive segmentation framework with the underlying automatic segmentation algorithm, an increased accuracy can be noted thanks to the manual surface steering driven by the user. Although this effect is not easily apprehended from the presented overall results, it should be noted that the user input is particularly relevant in challenging cases where the automatic segmentation is sub-optimal. By defining some segmentation quality threshold criteria from the observed intra-observer variability, it can be noted that 7/27 cases presented a Dice coefficient <85% using the automatic segmentation. However, only 4/27 and 2/27 cases were found to have Dice<85% in the interactive segmentation by both users. Similarly, for the mean absolute distance, 6/27 cases were found to have MAD > 3 mm using the automatic BEAS algorithm, while only 3/27 and 2/27 presented MAD > 3 mm in its interactive counterpart by both users respectively. This clearly points towards the added value of user input in challenging data where automatic segmentation might be suboptimal.

The proposed interactive segmentation approach offers an efficient and accurate alternative to full manual contouring. Indeed, interactiveBEAS leads to a 2 to 5-fold reduction of total analysis time, while keeping the surface segmentation error within the inter-observer variability range. Furthermore, the EDV extracted with the proposed method shows strong correlation and tight limits of agreement with respect to the reference value estimated by the two clinical experts. Despite user-dependent bias may be conserved, overall inter-observer variability is strongly reduced using interactiveBEAS. As a last remark, the computational efficiency of the underlying segmentation framework opens a path towards novel and improved user-interaction paradigms, as shown in the video available in the multimedia contents of the present manuscript.

Comparing interactiveBEAS with TurtleSeg, it is possible to observe that the EDV volume estimates present tighter limits of agreement with respect to the reference EDV values. Nonetheless, user-dependent bias are in part conserved with interactiveBEAS, which arises from the user input integrated in the segmentation algorithm to deform the LV surface provided by the automatic segmentation algorithm.

LV segmentation using interactiveBEAS was found to be the one providing the LV surfaces closer to the average contour of both clinical experts, although statistical significance was reached only for one user. Nonetheless, the performance of the proposed algorithm outranks TurtleSeg in the correct identification of the endocardial surface in RT3DE. Although EDV is an important volumetric cardiac

#### Table 2

Total analysis time for manual contouring and interactive segmentation for both experts ( $\mu \pm \sigma$  in s).

Manual contouring		interactiveBEAS		TurtleSeg	
User 1	User 2	User 1	User 2	User 1	User 2
$228.7\pm49.1$	$214.6 \pm 55.0$	$40.4\pm16.5$	$99.5\pm61.9$	$210.5\pm60.7$	$283.7\pm52.9$



**Fig. 6.** Segmentation results for two cases (three columns on the left and right respectively), visualized in two perpendicular long axis (*XZ* and *YZ* plane) and one short axis (*XY* plane). The different rows show the comparison between manual contouring (a), automatic segmentation (b) and interactive segmentations using the proposed method (c) and TurtleSeg (d). The two different users are represented by the contours red and green, for each segmentation approach respectively. The reference mean contour is illustrated in all images in yellow, whereas the automatic segmentation LV contour is represented in magenta. (For interpretation of references to color in this figure legend, the reader is referred to the web version of this article.)

index, not only the accurate estimation of this value is important. Indeed, several advanced processing algorithms, such as speckle tracking for cardiac strain assessment, require a rigorous spatial definition of the LV boundaries to be tracked during the cardiac cycle. Therefore, not only the correct LV volume should be recovered from the segmentation algorithm, but also the spatial position of the endocardial surface needs to be accurately resolved.

The results of the present study clearly show the inverse relationship between the degree of automation of the segmentation tool and its inter-observer variability, as well as total time of

#### Table 3

LV end-diastolic volume assessment using both automatic and interactive segmentation approaches compared to the estimated reference value (*R*: Pearson product-moment correlation coefficient; Bias: Bland-Altman bias; LOA: Bland-Altman limits of agreement).

	autoBEAS	interactiveBEAS		TurtleSeg	
		User 1	User 2	User 1	User 2
R	0.862	0.922	0.952	0.886	0.913
Bias ( $\mu$ , ml)	$-7.17^{*}$	$-10.99^{*}$	1.46	-5.33	$-14.86^{*}$
LOA ( $\pm 1.96\sigma$ , ml)	27.4	20.8†	16.8 <sup>†,‡</sup>	33.6	23.0

\* *p* < 0.05, paired *t*-test against zero.

<sup>†</sup> p < 0.05, f-test against TurtleSeg (user 1).

<sup>‡</sup> p < 0.05, f-test against autoBEAS.

## Table 4

Surface segmentation errors with respect to the estimated reference surface (MAD: mean absolute distance; HD: Hausdorf distance).

	autoBEAS	interactiveBEAS		TurtleSeg	
		User 1	User 2	User 1	User 2
Dice $(\mu + \sigma \%)$	$87.0\pm4.4$	$88.3 \pm 3.0^{*,**}$	$88.8 \pm 2.9^{\circ, \circ \circ}$	$88.1\pm2.7^*$	85.5 ± 3.4
MAD	$2.62\pm0.80$	$2.39 \pm 0.44^{*,**}$	$2.33\pm0.42^*$	$2.47\pm0.43^{*}$	$2.97\pm0.58$
$(\mu \pm \sigma, mm)$ HD $(\mu \pm \sigma, mm)$	8.69 ± 3.11	$7.97 \pm 1.84$	$7.03 \pm 1.86^{*,**}$	$8.15\pm2.03$	$\textbf{8.89} \pm \textbf{1.82}$

\* p < 0.05, paired t-test against TurtleSeg (user 2).

<sup>\*\*</sup> *p* < 0.05, paired *t*-test against autoBEAS.

#### Table 5

LV end-diastolic volume inter-observer variability analysis for manual contouring and interactive segmentation (*R*: Pearson product-moment correlation coefficient; *E*: absolute unsigned volume difference between the experts).

	Manual contouring	interactiveBEAS	TurtleSeg	
$R \\ E(\mu \pm \sigma, ml)$	$\begin{array}{c} 0.829 \\ 16.6 \pm 14.3 \end{array}$	$\begin{array}{c} \textbf{0.908} \\ \textbf{13.8} \pm \textbf{8.8}^{\dagger} \end{array}$	$\begin{array}{c} 0.916\\ 19.0\pm13.0\end{array}$	

<sup>†</sup> p < 0.05, *f*-test against manual contouring.

#### Table 6

Surface segmentation inter-observer variability analysis for manual contouring and interactive segmentation (MAD: mean absolute distance; HD: Hausdorf distance).

	Manual contouring	interactiveBEAS	TurtleSeg	
Dice $(\mu \pm \sigma, \%)$ MAD $(\mu \pm \sigma, mm)$ HD $(\mu \pm \sigma, mm)$	$\begin{array}{c} 86.1 \pm 4.2 \\ 2.86 \pm 0.75 \\ 11.37 \pm 4.32 \end{array}$	$88.9 \pm 3.1^{++}$ 2.45 ± 0.55 <sup>+++</sup> 7.64 ± 2.48 <sup>++++</sup>	$\begin{array}{c} 85.4 \pm 3.4 \\ 2.84 \pm 0.57 \\ 10.07 \pm 2.94 \end{array}$	

p < 0.05, paired *t*-test against manual contouring.

\*\* *p* < 0.05, paired *t*-test against TurtleSeg.

#### Table 7

Segmentation performance comparison (BA: Bland-Altman analysis, R: Pearson product-moment correlation coefficient, # number of cases).

	# Exams	EDV (ml)		Dice coefficient	MAD (mm)
		$\overline{\text{BA}\left(\mu\pm1.96\sigma\right)}$	R	$(\mu \pm \sigma)$	$(\mu \pm \sigma)$
interactiveBEAS (user1)	27	$-10.99 \pm 20.8$	0.92	88.3±3.0	$2.39\pm0.44$
interactiveBEAS (user2)	27	$-1.46 \pm 16.8$	0.95	$88.8\pm2.9$	$2.33 \pm 0.42$
Angelini et al. [39]	10	$16.1 \pm 50.1$	0.63	Х	Х
Hansegaard et al. [40]	21	$-5.0 \pm 21$	0.91	Х	$2.2\pm0.56$
Leung et al. [41]	99	$1.47 \pm 35.4$	0.95	Х	$2.85\pm1.01$
Rajpoot et al. [42]	26	$-5.1\pm48.5$	Х	$82.0\pm 6.0$	$2.15\pm0.66$

analysis. Indeed, a gradual reduction in variability of the EDV can be observed, from the manual contouring to the TurtleSeg segmentation and to the proposed approach. However, our results also demonstrate that the amount of user input required by livewirebased segmentation of RT3DE data leads to a inter-observer subjectivity of the spatial position of the endocardial surface nearly as high as the manual segmentation process. This supports the key advantage offered by the proposed algorithm. Indeed, since the user starts from a competitive automatic segmentation result, he will only have to perform minor adjustments, which will only affect locally the segmented endocardial surface. This results in lower total analysis times and reduced inter-observer variability, two fundamental requirements for clinical usability of software tools.

It should be noted that in the current work the user was not able to control the degree and span of the geometrical regularizer. However, and since the optimal amount of geometrical smoothing is likely to be user-dependent, an online control for this term will be introduced in our software tool.

The segmentation performance offered by the proposed interactive framework is competitive when compared with the current state-of-the-art. Without the claim of being exhaustive, we present in Table 7 a comparison between interactiveBEAS and relevant algorithms previously presented. For completness sake, it should be referred that the surface distance errors presented in Hansegaard et al. refer to the entire cardiac cycle, while Rajpoot et al. have segmented multi-view fusion echocardiography data, which offers superior image quality when compared with clinical RT3DE.

As a last note, it is important to stress that TurtleSeg is a generic tool which has not been developped specifically for cardiac applications. Thus, the 3D surface fitting to contours introduced interactively by the user is suboptimal, since it does not take into account the LV anatomy. Similarly, the automatic plane suggestion tool requires some adaptation from the user, which might explain the differences in the total analysis time between both users. Therefore, the presented results for TurtleSeg should be taken as a lower bound of the performance of an interactive segmentation method based on the livewire paradigm. Despite this, the proposed interactive segmentation clearly provides a substantial increase in the automation of the LV boundary delineation task.

## 6. Study limitations

The most important limitation of the present work is the fact that only two clinical experts were included in the study. Indeed, significant inter-observer variability in manual delineation of LV contours in RT3DE is widely acknowledged and thus several experts are needed to properly assess the statistical relevance of differences found between methods. Thus the presented results should be carefully interpreted, while keeping in mind that additional validation is still required in a larger patient database and including more reference contours from clinical experts.

The ROI definition for geometrical regularization was based on empirical observations while developing the proposed interactive algorithm. Indeed, 1/3 of the average longitudinal arclength arised as an optimal balance between a too locally restricted user deformation of the segmented LV (cf. Fig. 3B) and a too global surface over-smoothing effect. However, it is acknowledged that the optimal ROI definition might be user dependent. Therefore, future work will focus the inclusion of a user-defined span of the geometrical regularization, thus adapting the algorithm to the preference of the analyzing physician.

A note should be also addressed to the underlying external forces of the algorithms under comparison in the present work. While the automatic segmentation method used as starting point in the proposed interactive segmentation algorithm uses localized region-based forces, the livewire-based algorithm used for comparison purposes employs a gradient-based image force. Despite this may influence the accuracy of the features extracted from the image data, we would like to stress that in the present manuscript we mostly focus on how the information input by the user can be used to steer the segmentation result, by comparing the proposed algorithm with a livewire-based interactive segmentation paradigm. Furthermore, no additional high-level priors, such as statistical shape models or ultrasound-physics based terms, are employed in the automatic segmentation, whereas the local means separation can be seen as higher scale equivalent of the image gradient along the normal direction to the LV surface.

# 7. Conclusion

The proposed interactive framework allows to explore a synergistic balance between fully automatic segmentation and user-based interactive segmentation. By only allowing local user-based steering of the automatically segmented surface, the proposed method is able to significantly increase its accuracy while keeping its robustness to the subjectivity introduced by the user. To improve the fluency of the interaction, we introduced a geometrical regularization term allowing to reduce the number of points necessary to efficiently modify the contour. Furthermore, and thanks to the real-time nature of the underlying segmentation framework (BEAS), the interaction with the segmented surface provides immediate feedback on the modifications introduced by the user. This

reduces the additional overhead on the total analysis time and makes the interaction process less cumbersome.

# Acknowledgement

This work was supported by FCT, Portuguese Ministry of Science, Technology and Higher Education, through the grant SFRH/BD/62230/2009, and by the Rhône-Alpes region, through a Explora'Doc and a Accueil'Doc scholarship.

# Appendix – User-interaction energy derivation

Taking the proposed functional for surface deformation via user-based online steering, it is possible to directly express its minimization wrt. to the surface parameters by computing the derivatives of (5) wrt. each B-spline coefficient  $c[\mathbf{k}_i]$ :

$$\frac{\partial E_{user}}{\partial c[k_i]} = \int \int_{\Gamma} \frac{\partial \delta \left( (\theta, \varphi) - (\theta_{user}, \varphi_{user}) \right)}{\partial c[k_i]} (\psi(\theta, \varphi) - \rho_{user})^2 + \delta((\theta, \varphi) - (\theta_{user}, \varphi_{user})) \frac{\partial (\psi(\theta, \varphi) - \rho_{user})^2}{\partial c[k_i]} d\theta d\varphi.$$
(7.1)

Since  $\partial \delta ((\theta, \varphi) - (\theta_{user}, \varphi_{user})) / \partial c[k_i] = 0$ , the previous equation can be simplified to:

$$\frac{\partial E_{user}}{\partial c[\mathbf{k}_{i}]} = 2 \int \int_{\Gamma} \delta\left((\theta, \varphi) - (\theta_{user}, \varphi_{user})\right) (\psi(\theta, \varphi) - \rho_{user}) \\ \times \frac{\partial \psi(\theta, \varphi)}{\partial c[\mathbf{k}_{i}]} d\theta d\varphi.$$
(7.2)

Considering from (2) that:

$$\frac{\partial \psi(\theta, \varphi)}{\partial c[\mathbf{k}_i]} = \beta^d \left( \frac{(\theta, \varphi)}{h} - \mathbf{k}_i \right), \tag{7.3}$$

developping 7.2 yields Eq. (6).

# Appendix – Integrating the proposed geometrical regularizer in BEAS

The derivation of Eq. (10) is obtained by first noting that the surface diffusion flow corresponds to a particular instance of the more general case of the Willmore surfaces [37]. If a surface  $\Gamma$  is expressed as an implicit function  $\phi(\mathbf{x})$  (*i.e.* as a level set) defined over the domain  $\Omega$ , it has been shown [38] that it corresponds to a Willmore surface if it minimizes the following energy:

$$E_W = \frac{1}{2} \int_{\Omega} \kappa_M^2(\mathbf{x}) ||\nabla_{\phi}(\mathbf{x})|| \delta_{\phi}(\mathbf{x}) d\mathbf{x}, \qquad (7.4)$$

where  $\delta_{\phi}(\mathbf{x}) = \delta(\phi(\mathbf{x}))$  and  $\delta$  is the Dirac delta function.

The minimization of  $E_W$  can be done by minimizing its first variation w.r.t.  $\phi$ , which has been shown to be [38]:

$$\frac{\partial E_W}{\partial \phi} = ||\nabla_{\phi}(\mathbf{x})|| \left(\Delta \kappa_M(\mathbf{x}) + \kappa_M^2(\mathbf{x}) \left(||S||^2 - \frac{1}{2}\kappa_M(\mathbf{x})\right)\right) \delta_{\phi}(\mathbf{x}), \quad (7.5)$$

where *S* denotes the shape operator applied in the surface  $\Gamma$ .

In this framework, the surface diffusion flow is obtained as a simplification of the Willmore surfaces by setting the second term of the sum to zero [37], which yields:

$$\frac{\partial E_{SDF}}{\partial \phi} = ||\nabla_{\phi}(\mathbf{x})|| \Delta \kappa_{M}(\mathbf{x}) \delta_{\phi}(\mathbf{x}).$$
(7.6)

Since we are considering the B-spline representation of the surface, the flow has to be expressed as the variation of  $E_{SDF}$  w.r.t the B-spline coefficients  $c[\mathbf{k}]$ . In this case, the following result holds [23,28]:

$$\frac{\partial E_{SDF}}{\partial c[\mathbf{k}_i]} = \int_{\Omega} \frac{\partial E_{SDF}}{\partial \phi} \frac{\partial \phi}{\partial c[\mathbf{k}_i]} d\mathbf{x}.$$
(7.7)

From (7.6), we thus obtain:

$$\frac{\partial E_{\text{SDF}}}{\partial c[\mathbf{k}_i]} = \int_{\Omega} ||\nabla_{\phi}(\mathbf{x})|| \Delta \kappa_M(\mathbf{x}) \frac{\partial \phi}{\partial c[\mathbf{k}_i]} \delta_{\phi}(\mathbf{x}) d\mathbf{x}.$$
(7.8)

The implicit and explicit expressions of the surface integral of a 3D function *f* are related through the following equation [23]:

$$\int_{\Omega} f(\mathbf{x}) \delta_{\phi}(\mathbf{x}) d\mathbf{x} = \int_{\Gamma} f(\mathbf{x}^*) d\mathbf{x}^*.$$
(7.9)

where  $f(\mathbf{x}^*)$  is the restriction of f over the surface  $\Gamma$ . Applying this result to Eq. (7.8) we have:

$$\frac{\partial E_{SDF}}{\partial c[\mathbf{k}_i]} = \int_{\Gamma} ||\nabla_{\phi}(\mathbf{x}^*)|| \Delta \kappa_M(\mathbf{x}^*) \frac{\partial \phi}{\partial c[\mathbf{k}_i]} d\mathbf{x}^*.$$
(7.10)

Noting moreover that:

$$\frac{\partial \psi(\theta,\varphi)}{\partial c[\mathbf{k}_{i}]} = \beta^{d} \left( \frac{(\theta,\varphi)}{h} - \mathbf{k}_{i} \right), \tag{7.11}$$

finally yields Eq. (10).

# Appendix A. Supplementary Data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.compmedimag. 2013.10.002.

## References

- [1] Olabarriaga S, Smeulders A. Interaction in the segmentation of medical images: a survey. Medical Image Analysis 2001;5(2):127–42.
- [2] Heckel F, Konrad O, Hahn HK, Peitgen H-O. Interactive 3D medical image segmentation with energy-minimizing implicit functions. Computers & Graphics 2011;35(2):275–87.
- [3] Karasev P, Kolesov I, Chudy K, Tannenbaum A, Muller G, Xerogeanes J. Interactive MRI segmentation with controlled active vision. In: 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC). 2011, p. 2293–8.
- [4] de Bruin P, Dercksen V, Post F, Vossepoel A, Streekstra G, Vos F. Interactive 3D segmentation using connected orthogonal contours. Computers in Biology and Medicine 2005;35(4):329–46.
- [5] Kang Y, Engelke K, Kalender WA. Interactive 3D editing tools for image segmentation. Medical Image Analysis 2004;8(1):35–46.
- [6] Young AA, Cowan BR, Thrupp SF, Hedley WJ, Dell'Italia LJ. Left ventricular mass and volume: fast calculation with guide-point modeling on MR images1. Radiology 2000;216(2):597–602.
- [7] Falc ao A, Udupa J, Miyazawa F. An ultra-fast user-steered image segmentation paradigm: live wire on the fly. IEEE Transactions on Medical Imaging 2000;19(1):55–62.
- [8] Falc ao A, Udupa J. A 3D generalization of user-steered live-wire segmentation. Medical Image Analysis 2000;4(4):389–402.
- [9] Hamarneh G, Yang J, Mcintosh C, Langille M. 3D live-wire-based semiautomatic segmentation of medical images. In: Proceedings of SPIE Medical Imaging: Image Processing, vol. 5747. 2005. p. 1597–603.
- [10] Lu K, Higgins W. Interactive segmentation based on the live wire for 3D CT chest image analysis. International Journal of Computer Assisted Radiology and Surgery 2007;2:151–67.
- [11] Poon M, Hamarneh G, Abugharbieh R. Efficient interactive 3D livewire segmentation of complex objects with arbitrary topology. Computerized Medical Imaging and Graphics 2008;32(8):639–50.
- [12] Grady L. Minimal surfaces extend shortest path segmentation methods to 3D. IEEE Transactions on Pattern Analysis and Machine Intelligence 2010;32(2):321–34.
- [13] Boykov Y, Veksler O, Zabih R. Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 2001;23:2001.

- [14] Boykov Y, Funka-Lea G. Graph cuts and efficient n-d image segmentation. International Journal of Computer Vision 2006;70:109–31.
- [15] Boykov Y, Kolmogorov V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence 2004;26(9):1124–37.
- [16] Kass M, Witkin A, Terzopoulos D. Snakes: active contour models. International Journal Of Computer Vision 1988;1:321–31.
- [17] Yushkevich PA, Piven J, Hazlett HC, Smith RG, Ho S, Gee JC, Gerig G. Userguided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability. NeuroImage 2006;31(3):1116–28.
- [18] Delgado-Gonzalo R, Chenouard N, Unser M. Fast parametric snakes for 3D microscopy. In: 9th IEEE International Symposium on Biomedical Imaging (ISBI). 2012. p. 852–5.
- [19] Liang J, McInerney T, Terzopoulos D. United snakes. Medical Image Analysis 2006;10(2):215–33.
- [20] Eyiyurekli M, Breen D. Interactive free-form level-set surface-editing operators. Computers & Graphics 2010;34(5):621–38.
- [21] Jalba A, van der Laan W, Roerdink JBTM. Fast sparse level sets on graphics hardware. IEEE Transactions on Visualization and Computer Graphics 2013;19(1):30-44, doi:10.1109/TVCG.2012.69.
- [22] Roberts M, Packer J, Sousa MC, Mitchell JR. A work-efficient GPU algorithm for level set segmentation. In: Proceedings of the Conference on High Performance Graphics, HPG '10, Eurographics Association. 2010. p. 123–32.
- [23] Barbosa D, Dietenbeck T, Schaerer J, D'hooge J, Friboulet D, Bernard O. B-spline explicit active surfaces: An efficient framework for real-time 3D region-based segmentation. IEEE Transactions on Image Processing 2012;1(1):241–51.
- [24] Barbosa D, Heyde B, Dietenbeck T, Houle H, Friboulet D, D'hooge J, Bernard O. Quantification of left ventricular volume and global function using a fast automated segmentation tool: validation in a clinical setting. The International Journal of Cardiovascular Imaging 2012;29(2):309–16.
- [25] Barbosa D, Dietenbeck T, Heyde B, Houle H, Friboulet D, D'hooge J, Bernard O. Fast and fully automatic 3D echocardiographic segmentation using B-spline explicit active surfaces: feasibility study and validation in a clinical setting. Ultrasound in Medicine and Biology 2013;39(1):89–101.
- [26] Duan Q, Angelini ED, Laine AF. Real-time segmentation by active geometric functions. Computer Methods and Programs in Biomedicine 2010;98(3): 223–30.
- [27] Unser M. Splines: a perfect fit for signal and image processing. IEEE Signal Processing Magazine 1999;16(6):22–38.
- [28] Bernard O, Friboulet D, Thévenaz P, Unser M. Variational B-spline level-set: a linear filtering approach for fast deformable model evolution. IEEE Transactions on Image Processing 2009;18:1179–91.

- [29] Gelas A, Bernard O, Friboulet D, Prost R. Compactly supported radial basis functions based collocation method for level-set evolution in image segmentation. IEEE Transactions on Image Processing 2007;16(7):1873–87, doi:10.1109/TIP.2007.898969.
- [30] Chan T, Vese L. Active contours without edges. IEEE Transactions on Image Processing 2001;10:266–77.
- [31] Yezzi A, Tsai A, Willsky A. A fully global approach to image segmentation via coupled curve evolution equations. Journal of Visual Communication and Image Representation 2002;13:195–216.
- [32] Lankton S, Tannenbaum A. Localizing region-based active contours. IEEE Transactions on Image Processing 2008;17(11):2029–39.
- [33] Delingette H, Montagnat J. Shape and topology constraints on parametric active contours. Computer Vision and Image Understanding 2001;83(2):140–71.
- [34] Claus P, Choi HF, D'hooge J, Rademakers FE. On the calculation of principle curvatures of the left-ventricular surfaces. In: 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2008. EMBS 2008. 2008. p. 961–4.
- [35] Top A, Hamarneh G, Abugharbieh R. Active learning for interactive 3D image segmentation. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI), Vol. 6893 of LNCS. 2011. p. 603–10.
- [36] Top A, Hamarneh G, Abugharbieh R. Spotlight: automated confidence-based user guidance for increasing efficiency in interactive 3D image segmentation. In: Medical Image Computing and Computer-Assisted Intervention Workshop on Medical Computer Vision (MICCAI MCV). 2010. p. 204–13.
- [37] Xu G, Zhang Q. G<sup>2</sup> surface modeling using minimal mean-curvature-variation flow. Computer-Aided Design 2007;39(5):342–51.
- [38] Droske M, Rumpf M. A level set formulation for Willmore flow. Interfaces and Free Boundaries 2004;6(3):361–78.
- [39] Angelini E, Homma S, Pearson G, Holmes J, Laine A. Segmentation of real-time three-dimensional ultrasound for quantification of ventricular function: a clinical study on right and left ventricles. Ultrasound in Medicine and Biology 2005;31(9):1143–58.
- [40] Hansegard J, Orderud F, Rabben S. Real-time active shape models for segmentation of 3D cardiac ultrasound. In: Kropatsch W, Kampel M, Hanbury A, editors. Computer Analysis of Images and Patterns. Springer Berlin / Heidelberg: Vol. 4673 of Lecture Notes in Computer Science; 2007. p. 157–64.
- [41] Leung K, van Stralen M, van Burken G, de Jong N, Bosch J. Automatic active appearance model segmentation of 3D echocardiograms. In: Proc. of the IEEE International Symposium on Biomedical Imaging (ISBI2010). 2010. p. 320–3.
- [42] Rajpoot K, Grau V, Noble JA, Becher H, Szmigielski C. The evaluation of single-view and multi-view fusion 3D echocardiography using image-driven segmentation and tracking. Medical Image Analysis 2011;15(4):514–28.