Real-time Tracking of the Left Ventricle in 3D Echocardiography Using a State Estimation Approach

Fredrik Orderud[1], Jøger Hansgård[2], and Stein I. Rabben[3]

[1] Norwegian University of Science and Technology, Norway
`fredrik.orderud@idi.ntnu.no`
[2] University of Oslo, Norway `jogerh@ifi.uio.no`
[3] GE Vingmed Ultrasound, Norway `stein.rabben@med.ge.com`

Abstract. In this paper we present a framework for real-time tracking of deformable contours in volumetric datasets. The framework supports composite deformation models, controlled by parameters for contour shape in addition to global pose. Tracking is performed in a sequential state estimation fashion, using an extended Kalman filter, with measurement processing in information space to effectively predict and update contour deformations in real-time. A deformable B-spline surface coupled with a global pose transform is used to model shape changes of the left ventricle of the heart.

Successful tracking of global motion and local shape changes without user intervention is demonstrated on a dataset consisting of 21 3D echocardiography recordings. Real-time tracking using the proposed approach requires a modest CPU load of 13% on a modern computer. The segmented volumes compare to a semi-automatic segmentation tool with 95% limits of agreement in the interval 4.1 ± 24.6 ml ($r = 0.92$).

1 Introduction

The emergence of volumetric image acquisition within the field of medical imaging has attracted a lot of scientific interest over the last years. In a recent survey, Noble *et al.* [1] presented a review of the most significant attempts for automatic segmentation within the field of ultrasonics. However, all of these attempts are limited to being used as postprocessing tools, due to extensive processing requirements, even though volumetric acquisition may be performed in real-time with the latest generation of 3D ultrasound technology. Availability of technology for real-time tracking and segmentation in volumetric datasets would open up possibilities for instant feedback and diagnosis during data acquisition. Automatic tracking of the main chamber of the heart, the left ventricle (LV), would here serve as an excellent application.

Orderud [2] recently presented a tracking approach that allows for real-time tracking of rigid bodies in volumetric datasets. The approach treats the tracking problem as a state estimation problem, and uses an extended Kalman filter to

recursively track global pose parameters using a combination of state predictions and measurement updates. Experimental validation of LV tracking in 3D echocardiography, using a simple truncated ellipsoid model, was performed to demonstrate the feasibility of the approach.

This state estimation approach is based on prior work by Blake *et al.* [3, 4], who used a Kalman filter to track B-spline contours deformed by linear transforms within a model subspace referred to as *shape space*. Later, the same approach was applied to real-time LV tracking in 2D echocardiography by Jacob *et al.* [5–7]. All these papers did, however, lack a separation between global pose and local shape. They were also restricted to linear deformations, such as principal component analysis deformation models, and to tracking in 2D datasets.

This paper extends [2] to support contours with composite deformation models, consisting of both local shape changes and global pose. We also propose to use a 3D B-spline surface model to circumvent the limitations of rigid ellipsoidal models for LV tracking. This model is coupled with a global pose transform, and successful LV tracking in 3D echocardiography is demonstrated.

This model somewhat resembles the deformable superquadrics model [8], and a 3D active shape model [9], but is spline-based, and allows for free-form shape deformations [10] by letting the control points move independently perpendicular to the surface.

2 Deformable Contour

The tracking framework is based upon a contour deformation model $\mathbf{T}$, which is decomposed into local deformations and global transformations. Local shape deformations are used to alter contour shape, by deforming points on a shape template $\mathbf{p}_0$ into intermediate contour shape points $\mathbf{p}_l$, using a local shape deformation model $\mathbf{T}_l$ with local state vector $\mathbf{x}_l$ as parameters:

$$\mathbf{p}_l = \mathbf{T}_l(\mathbf{p}_0, \mathbf{x}_l) \ . \tag{1}$$

The intermediate contour shape points $\mathbf{p}_l$ are subsequently transformed into final contour points $\mathbf{p}$, using a global pose transformation model $\mathbf{T}_g$ with global state vector $\mathbf{x}_g$ as parameters:

$$\mathbf{p} = \mathbf{T}_g(\mathbf{p}_l, \mathbf{x}_g) \ . \tag{2}$$

A composite deformation model $\mathbf{T}$ is then constructed by combining local and global deformations of shape template points into a joint model, as shown in Fig. 1. This yields a composite state vector $\mathbf{x} \equiv \left[\mathbf{x}_g^T, \mathbf{x}_l^T\right]^T$ consisting of N_g global and N_l local deformation parameters. Calculation and propagation of associated contour normals $\mathbf{n}$ through $\mathbf{T}$ must also be performed, since they are required to specify the search direction for the later edge-detection.

This separation between local and global deformations is intended to ease modeling, since changes in shape are often parameterized differently compared to deformations associated with global position, size and orientation. It also
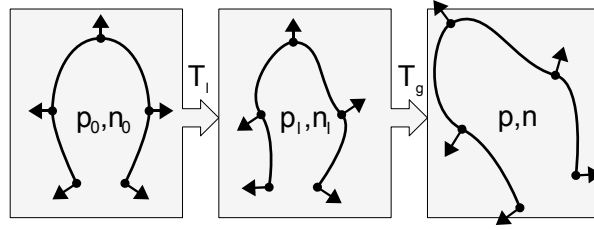
Fig. 1. Overview over the contour deformation and transformation process: The shape template $\mathbf{p}_0,\mathbf{n}_0$ is first deformed locally into $\mathbf{p}_l,\mathbf{n}_l$, followed by a global transformation into the final contour $\mathbf{p},\mathbf{n}$.

puts very few restrictions on the allowable deformations, so a wide range of parameterized deformation models can be used, including nonlinear models, as opposed to previous *shape space* models that are limited to linear deformations.

3 Tracking Framework

The proposed tracking framework follows the processing chain of the Kalman filter, starting by predicting contour state using a motion model (section 3.1). The shape template is subsequently deformed to form a predicted contour, and associated normal vectors and state-space Jacobi matrices are computed (section 3.2). Edge detection is then performed locally along the normal vectors. Finally, all measurement information is assimilated in information space [11], and combined with the predicted contour to compute an updated contour state estimate (section 3.3). Figure 2 shows an overview over these steps.
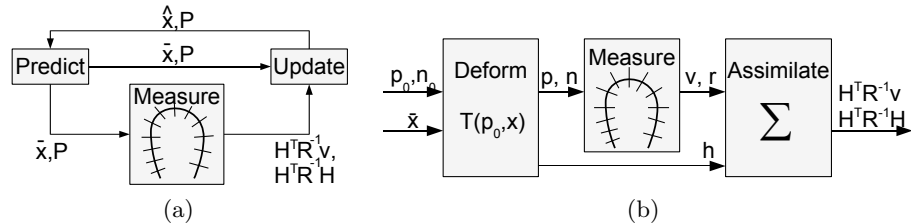

Fig. 2. (a) Overview of the tracking scheme. (b) Illustration of how points on the shape template $\mathbf{p}_0$, $\mathbf{n}_0$ are first deformed using a predicted state $\bar{\mathbf{x}}$, yielding a deformed contour $\mathbf{p}$, $\mathbf{n}$ and measurement vector $\mathbf{h}$. Edges are then measured relative to the predicted contour, resulting in normal displacements v with associated measurement error variances r. The edge measurements are finally assimilated into an information vector and matrix for efficient processing.

3.1 Motion Model

Modeling of motion, in addition to position, is accomplished in the prediction stage of Kalman filter by augmenting the state vector to contain the last two successive state estimates. A *motion model* which predicts the state $\bar{\mathbf{x}}$ at timestep $k + 1$, is then expressed as:

$$\bar{\mathbf{x}}_{k+1} - \mathbf{x}_0 = \mathbf{A}_1(\hat{\mathbf{x}}_k - \mathbf{x}_0) + \mathbf{A}_2(\hat{\mathbf{x}}_{k-1} - \mathbf{x}_0) \ , \tag{3}$$

where $\hat{\mathbf{x}}_k$ is the estimated state from timestep k. Tuning of properties like damping and regularization towards the mean state $\mathbf{x}_0$ for all deformation parameters can then be accomplished by adjusting the coefficients in matrices $\mathbf{A}_1$ and $\mathbf{A}_2$. Prediction uncertainty can similarly be adjusted by manipulating the process noise covariance matrix $\mathbf{B}_0$ used in the associated covariance update equation. The latter will then restrict the change rate of parameter values.

3.2 Edge Measurements

Processing of edge measurements using an extended Kalman filter [12] requires *state-space* Jacobi matrices to relate changes in contour point positions to changes in contour state. Separate Jacobi matrices for each contour point must therefore be calculated. The choice of composite deformations leads to state-space Jacobi matrices consisting of two separate parts, namely of global and local derivatives:

$$\frac{\partial \mathbf{T}(\mathbf{p}_0, \mathbf{x})_i}{\partial \mathbf{x}} \equiv \left[\frac{\partial \mathbf{T}_g(\mathbf{p}_l, \mathbf{x}_g)_i}{\partial \mathbf{x}_g}, \quad \sum_{n \in x,y,z} \frac{\partial \mathbf{T}_g(\mathbf{p}_l, \mathbf{x}_g)_i}{\partial \mathbf{p}_{l,n}} \frac{\partial \mathbf{T}_l(\mathbf{p}_0, \mathbf{x}_l)_n}{\partial \mathbf{x}_l} \right] \ . \tag{4}$$

The global Jacobian becomes a $3 \times N_g$ matrix, while the Jacobian for the local shape deformations becomes the product of a 3×3 matrix by a $3 \times N_l$ matrix using the chain rule for multivariate calculus.

Edge detection is performed by measuring the distance between points $\mathbf{p}_i$ on a predicted contour inferred from the motion model, and edges $\mathbf{p}_{obs,i}$ detected by searching in normal direction $\mathbf{n}_i$ of the contour surface. This type of edge detection is referred to as *normal displacement* [4], and is calculated as follows:

$$v_i = \mathbf{n}_i^T(\mathbf{p}_{obs,i} - \mathbf{p}_i) \ . \tag{5}$$

Each normal displacement measurement is coupled with a measurement noise value r_i that specifies the uncertainty associated with the edge, which may either be constant for all edges, or dependent on edge strength, or other measure of uncertainty.

Linearized measurement models [12], which are required in the Kalman filter for each edge measurement, are constructed by transforming the state-space Jacobi matrices the same way as the edge measurements, namely by projecting them onto the normal vector:

$$\mathbf{h}_i^T = \mathbf{n}_i^T \frac{\partial \mathbf{T}(\mathbf{p}_0, \mathbf{x})_i}{\partial \mathbf{x}} \ . \tag{6}$$

This yields a separate *measurement vector* $\mathbf{h}_i$ for each normal displacement measurement, to relate normal displacements to changes in contour state.

3.3 Measurement Assimilation and State Update

Contour tracking forms a special problem structure, since the number of measurements far exceeds the number of state dimensions. Ordinary Kalman gain calculations then becomes intractable, since they involve inverting matrices of dimensions equal to the number of measurements. An alternative approach is to assimilate measurements in *information space* prior to the state update step. This enables very efficient processing if we assume that measurements are independent. All measurement information can then be summed into an information vector and matrix of dimensions invariant to the number of measurements:

$$\mathbf{H}^T \mathbf{R}^{-1} \mathbf{v} = \sum_i \mathbf{h}_i r_i^{-1} v_i \tag{7}$$

$$\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} = \sum_i \mathbf{h}_i r_i^{-1} \mathbf{h}_i^T \ . \tag{8}$$

Measurements in information filter form require some alterations to the state update step in the Kalman filter. This can be accomplished by using the information filter formula [11] for the updated state estimate $\hat{\mathbf{x}}$ at timestep k:

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \hat{\mathbf{P}}_k \mathbf{H}^T \mathbf{R}^{-1} \mathbf{v}_k \ . \tag{9}$$

The updated error covariance matrix $\hat{\mathbf{P}}$ can similarly be calculated in information space to avoid inverting matrices with dimensions larger than the state dimension:

$$\hat{\mathbf{P}}_k^{-1} = \bar{\mathbf{P}}_k^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \ . \tag{10}$$

4 Experiments

We used a quadratic B-spline surface consisting of 24 control points arranged in a prolate spheroid grid as a LV model. Shape deformations were enabled by allowing the control points to move perpendicularly to the surface. This was combined with a global model for translation and scaling in three dimensions, as well as rotation around two of the axes. In total, this resulted in a deformation model consisting of 32 parameters. Edge detection was performed in the normal direction of approximately 450 contour points distributed evenly over the surface, using 30 samples spaced 1 mm apart. A simple edge detector based upon the transition criterion with variable height [13] is used to determine the position of the strongest edge along each normal. This detector assumes edges to form a transition in image intensity, from a dark cavity to a bright myocardium, and calculates the edge position that minimizes the sum of squared errors between a transition model and the data. Weak and outlier edges were automatically rejected to improve robustness.

4.1 Results

A collection of 21 apical 3D echocardiography recordings of adult patients, of which half were diagnosed with heart diseases, were used to validate the method.

The recordings were acquired with a Vivid 7 scanner (GE Vingmed Ultrasound, Norway), using sub-volume stitching to form a contiguous wide-angle recording of the LV. The exact same tracking configuration was used for all recordings, with an initial LV contour placed at a predefined position in the first frame. The tracking were then run for a couple of heartbeats to give the contour enough time to lock on to the LV. Tracking accuracy was evaluated by comparing the segmentations with the results from a custom-made semiautomatic segmentation tool (GE Vingmed Ultrasound), that facilitates manual editing. The reference segmentations were validated and, if needed, edited by an expert operator.

Figure 3 shows the overall volume correspondence throughout the cardiac cycle, for all frames in all of the 21 recordings. Bland-Altman analysis of the volume points in Fig. 3(a) yields a 4.1 ml bias, with 95% limits of agreement in the interval 4.1 ± 24.6 ml, and a strong correlation ($r = 0.92$). The average point to surface distance between the contour and the reference surfaces was 2.7 mm.

The volume curves were usually very similar in shape compared to the reference method. However, some per-recording bias was seen. This is illustrated in Fig. 3(b), where each volume curve is more or less parallel to the identity line, but have different offset. If we subtract the 'per-recording' bias from each volume curve, the limits of agreement is improved to ± 11.0 ml, indicating that much of the disagreement stems from the varying bias from recording to recording.

Figure 4 shows the intersection between the segmented contour and several slices throughout the volume in one of the recordings, as well as a plot of the volume of the segmented contour throughout the cardiac cycle.
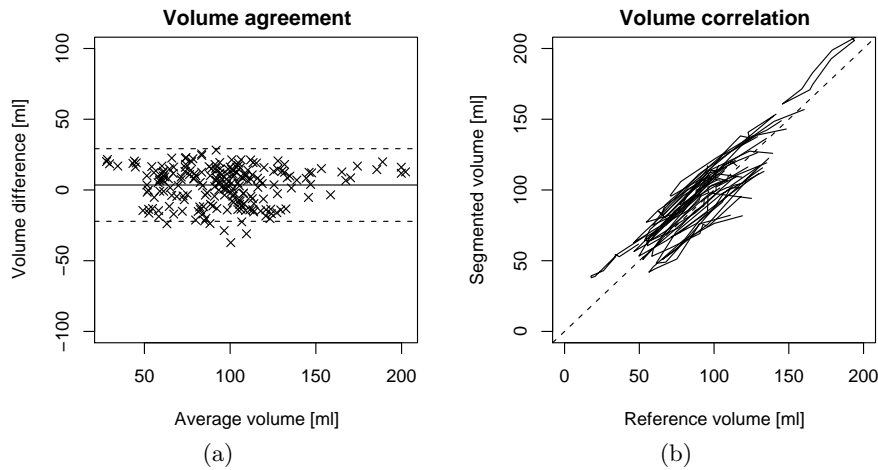

Fig. 3. Bland-Altman plot for the overall correspondence between automatically real-time segmented volumes and the reference (a), along with the mean volume difference (solid) and 95% limits of agreement (dashed). Associated volume correlation plot (b), where segmented volume curves from each recording is compared to the reference and shown as separate lines, along with the identity line (dashed).
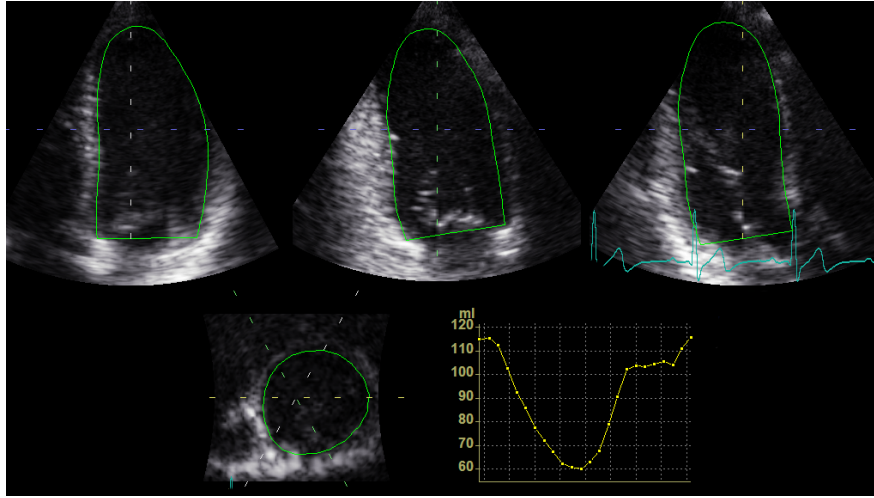
Fig. 4. Example of successful real-time tracking, showing intersections between the segmented contour and several slices through the volume, as well as a plot of the volume of the segmented contour for one cardiac cycle.

The CPU load required to maintain real-time segmentation at 25 frames per second (fps) was approximately 13% when visualization was disabled[4].

5 Discussion and Conclusion

A novel framework for real-time tracking of deformable contours in volumetric datasets using a state estimation approach has been proposed. It extends prior work [2] by enabling tracking of contours with both local and global modes of deformation, both of which may be nonlinear. Compared to traditional deformable model based segmentation methods, the non-iterative Kalman filter algorithm leads to outstanding computational performance.

The feasibility of real-time LV tracking in 3D echocardiography has been demonstrated by successful tracking and segmentation in 21 recordings using a quadratic B-spline model coupled to a global pose transform. This represents a significant improvement over [2], that solely tracked chamber position, orientation and size, using an ellipsoid model, and were thus inherently unable of segmenting shape changes of the LV. The framework is well suited for rapid analysis of LV volumes and global function, since it operates without user interaction. Other applications, like patient monitoring, and automatic initialization of other methods are also possible.

Tracking accuracy is primarily limited by the difficulty of edge detection in echocardiography recordings, which suffers from inherently poor image quality.

---

[4] The experiments were performed using a C++ implementation on a 2.16 GHz Intel Core 2 duo processor.

It is in fact difficult, even for experts, to accurately determine the endocardial border in such recordings. Replacement of the simplistic transition criterion with a more advanced edge-detector, specifically designed for the ultrasound modality, is therefore believed to give better tracking accuracy.

References

1. Noble, J.A., Boukerroui, D.: Ultrasound image segmentation: A survey. Medical Imaging, IEEE Transactions on **25**(8) (2006) 987–1010
2. Orderud, F.: A framework for real-time left ventricular tracking in 3D+T echocardiography, using nonlinear deformable contours and kalman filter based tracking. In: Computers in Cardiology. (2006)
3. Blake, A., Curwen, R., Zisserman, A.: A framework for spatiotemporal control in the tracking of visual contours. International Journal of Computer Vision **11**(2) (October 1993) 127–145
4. Blake, A., Isard, M.: Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion. Springer-Verlag New York, Inc., Secaucus, NJ, USA (1998)
5. Jacob, G., Alison Noble, J., Mulet-Parada, M., Blake, A.: Evaluating a robust contour tracker on echocardiographic sequences. Medical Image Analysis **3**(1) (March 1999) 63–75
6. Jacob, G., Noble, J.A., Kelion, A.D., Banning, A.P.: Quantitative regional analysis of myocardial wall motion. Ultrasound in Medicine & Biology **27**(6) (June 2001) 773–784
7. Jacob, G., Noble, J.A., Behrenbruch, C., Kelion, A.D., Banning, A.P.: A shape-space-based approach to tracking myocardial borders and quantifying regional left-ventricular function applied in echocardiography. Medical Imaging, IEEE Transactions on **21**(3) (2002) 226–238
8. Park, J., Metaxas, D., Young, A.A., Axel, L.: Deformable models with parameter functions for cardiac motion analysis from tagged MRI data. Medical Imaging, IEEE Transactions on **15**(3) (June 1996) 278–289
9. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models - Their training and application. Computer Vision and Image Understanding **61**(1) (January 1995) 38–59
10. Rueckert, D., Sonoda, L.I., Hayes, C., Hill, D.L.G., Leach, M.O., Hawkes, D.J.: Nonrigid registration using free-form deformations: application to breast MR images. Medical Imaging, IEEE Transactions on **18**(8) (1999) 712–721
11. Comaniciu, D., Zhou, X.S., Krishnan, S.: Robust real-time myocardial border tracking for echocardiography: An information fusion approach. Medical Imaging, IEEE Transactions on **23**(7) (2004) 849–860
12. Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation. Wiley-Interscience (2001)
13. Rabben, S.I., Torp, A.H., Støylen, A., Slørdahl, S., Bjørnstad, K., Haugen, B.O., Angelsen, B.: Semiautomatic contour detection in ultrasound M-mode images. Ultrasound in Med. & Biol. **26**(2) (Feb. 2000) 287–296