34th Annual International Conference of the IEEE EMBS
San Diego, California USA, 28 August - 1 September, 2012

Towards Localizing on the Surface of the Beating Heart

Nathan A. Wood, Tian Yu Tommy Liu, Kevin Waugh, Marco A. Zenati, and Cameron N. Riviere

***Abstract*— This paper presents preliminary work toward localizing on a surface which undergoes periodic deformation, as an aspect of research on HeartLander, a miniature epicardial crawling robot. Using only position measurements from the robot, the aim of this work is to use the nonuniform movements of the heart as features to aid in localization. Using a particle filter, with motion and observation models which accurately model the robotic system, registration and localization parameters can be quickly and accurately identified. The presented framework is demonstrated in simulation on dynamic 2-D models which approximate the deformation of the surface of the heart.**

I. Introduction

The continued rise in the appeal of minimally invasive cardiac therapies stems largely from the reduction in patient morbidity. Standard cardiac surgeries require invasive sternotomies or thoracotomies and result in longer patient stays and increased risk of infections. While minimally invasive methods provide many benefits to the patient, the instruments and access points used pose significant technological challenges [1]. Providing visual feedback to the clinician is one of these challenges because the small port-like incisions used do not permit line of sight to the operation field. This challenge has previously been overcome by using real time medical imaging such as fluoroscopy [2], magnetic resonance imaging (MRI) [3], or ultrasound [4]. Another method, image-guided surgery, is often used in robotic interventions [5], [6].

In image-guided surgery, 3-dimensional maps of the operating field are constructed from pre-operative medical images and provide a virtual view to the clinician. The surgical device, which is often tracked using an electromagnetic position sensor, is localized within the map and displayed in the visualization. Registration between the map and the operating site must also be found so that desired motion in the map frame can be translated to motion in the real world. This framework is closely tied to robotic interventions because in order to plan and act intelligently, the robot must possess a map of the environment and know where it is in that environment.

Generally, the maps used in image guided surgery are static. Although the heart undergoes periodic deformations of up to 30 mm due to the heartbeat and respiration [7], treating the heart as a dynamic body poses significant challenges for systems which move freely in the cardiothoracic cavity or inside the heart due to changing contact constraints. If, however, a robot is constrained to the surface of the heart, the dynamic nature of the heart may be used to improve localization and registration.

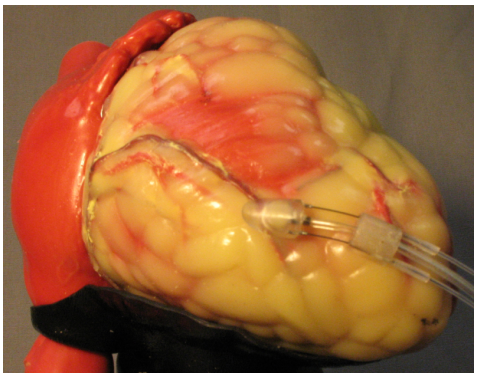


Fig. 1. The HeartLander robot.

HeartLander, shown in Fig. 1, is a miniature mobile robot which adheres to and moves across the surface of the heart to provide therapies in a minimally invasive manner. Access to the heart is gained via a subxiphoid skin incision and an incision in the pericardium. The robot adheres to the surface using suction, and moves by alternately extending and retracting drive wires, which controls the distance between the body sections, and alternating suction. Previous work has demonstrated the ability to access, locomote, and accurately reach targets in live animal testing [8]; however, the current methods used for localizing on the surface of the heart use several approximations which limit accuracy.

The current system uses a static heart model generated from pre-operative CT images, and because of this, position measurements, which come from a 6-degree-of-freedom electromagnetic tracking sensor (microBIRD, Ascension Technology) embedded in the front foot of the robot, are filtered to remove the periodic motion due to the physiological cycles. The filtered position measurement is treated as the position of the robot on the surface of the static heart model. The transformation between the map frame and measurement frame is found using markers placed on the chest wall which are identified in each frame.

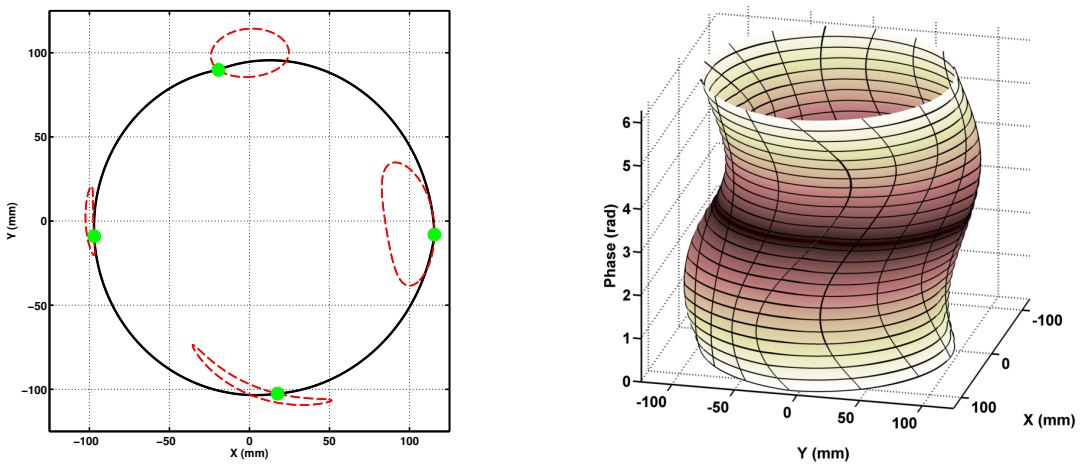
This work uses simplified 2-dimensional maps as a proof of concept for improving localization and registration accuracy by treating periodic deformations as features which yield information about the current robot position on the

This work was supported in part by the U.S. National Institutes of Health under Grant nos. R01 HL078839 and R01 HL105911.

N. A. Wood, T. Y. T. Liu, and C. N. Riviere are with The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213 {nwood@andrew, tianyul@andrew, camr@ri}.cmu.edu

K. Waugh is with the Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213 waugh@cs.cmu.edu

M. A. Zenati is with the BHS Department of Cardiothoracic Surgery, Harvard Medical School, West Roxbury, Massachusetts 02132



(a) (b)

Fig. 2. Example 2D surface generation. (a) Four anchor points, shown in green, follow the randomly-generated periodic motion, shown in red. (b) Visualizing the randomly-generated 2D surface in 3D where the vertical axis is phase. The 2D surface at each phase is the cross section of the 3D surface.

heart. The work presented uses a particle filter to estimate localization and registration parameters, is demonstrated in simulation on surfaces which approximate the surface of the heart, and is the initial step towards developing the methods for implementation on full 3-dimensional models.

II. Methods

A. Surface Generation

The work presented relies on possessing complete maps which describe the periodic motion of a surface. For our purposes, a map of a surface takes the form:

$$M = [x(\phi), y(\phi), \vec{n}(\phi)], \tag{1}$$

where $\phi \in (0, 2\pi]$ is the phase, x and y are Cartesian coordinates in map frame, and $\vec{n}$ are the surface normals.

In order to develop and test the following methods used for localizing on such surfaces, random 2D periodically deforming surfaces were generated in the following manner. The map of each surface is defined by a discrete number of points whose base shape is a circle, with a radius defined as r_{base}. The motion of the surface is set by randomly assigning periodic motion to a specified number of equally spaced anchor points. The motion of each of the anchor points is defined in polar coordinate as:

$$r_i(\phi) = r_{\text{base}} + \sum_{n=1}^{H_r} A_{i_n} \cos\left(n\phi + \alpha_{i_n}\right) \tag{2}$$

$$\theta_i(\phi) = \theta_{\text{base}_i} + \sum_{n=1}^{H_\theta} B_{i_n} \cos\left(n\phi + \beta_{i_n}\right), \tag{3}$$

where ϕ is the phase, θ_{base_i} are the anchor points base angular position which are equally spaced over $(0, 2\pi]$, H_r and H_θ are the number of harmonics in each Fourier series, and A_{i_n}, α_{i_n}, B_{i_n}, and β_{i_n} are randomly-generated Fourier series parameters. With the motion of the anchor points defined, the motion of all points in between the anchor points are then linearly interpolated in polar coordinates from the adjacent anchor points. The polar coordinates are transformed to Cartesian coordinates and surface normal are calculated and stored in the map.

An example of a randomly-generated 2D surface which undergoes periodic deformation is shown in Fig. 2(a). This map was generated using four anchor points, shown in green, whose periodic motion is depicted as dashed red lines. The same 2D surface is shown in 3D in Fig. 2(b). The vertical axis in the 3D plot is phase, where the 2D surface at a particular phase is then the cross section of the 3D surface.

B. Simulated System

The simulated system is represented by a given map, M, and the following state vector:

$$s_t = \begin{bmatrix} x_m & y_m & \phi & \Delta x & \Delta y & \Delta\theta \end{bmatrix}^T, \tag{4}$$

where x_m and y_m are the location of the robot on the surface in map coordinates, ϕ is the current phase, and Δx, Δy and $\Delta\theta$ are registration parameters defining how the map is translated and rotated in world coordinates. Using this representation, the location of the robot in the world frame is then:

$$\begin{bmatrix} x_w \\ y_w \end{bmatrix} = \begin{bmatrix} \cos\Delta\theta & -\sin\Delta\theta & \Delta x \\ \sin\Delta\theta & \cos\Delta\theta & \Delta y \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} \tag{5}$$

The phase of the system is advanced by:

$$\phi_t = \phi_{t-1} + \omega dt, \tag{6}$$

where the velocity, ω, is assumed to be constant.

Control inputs to the robot, u_t, move the robot along the surface a distance $|u_t|$ counterclockwise for negative inputs, and clockwise for positive inputs. Using this framework, we wish to estimate the current state vector, s_t, using a particle filter.

C. Particle Filter

This section gives a description of the particle filter algorithm implemented in this work. The particle filter is a nonparametric Bayes filter which represents the posterior distribution by a set of random samples, or particles, drawn from the posterior. Each particle is a hypothesis of the the true state of the system, where the likelihood for each state hypothesis is proportional to its Bayes posterior, and the set of particles represents the the distribution over possible states. A more in-depth treatment of the algorithm and related topics can be found in [9].

1) Particle Initialization: Initially it is assumed that the only information available at initialization is a single measurement of the pose of the robot in the world frame. Although this measurement can not be leveraged to gain any knowledge about where on the surface the robot currently is, one can use this measurement to restrict the space of registration parameters. At initialization, a fixed number of particles, N, is generated, randomly distributed over the surface and phase space. Using the known location of the particles on the map, the registration parameters are instantiated such that each particle, when transformed to world coordinates, would produce the initial measurement.

2) Motion Model: Incorporation of the control inputs in the state transition distribution is achieved through use of a robot motion model. As previously described, the control input u_t move the robot along the surface a distance $|u_t|$ clockwise or counterclockwise depending on the sign of u_t. In order to return a sample from the distribution $p(s_t|u_t, s^i_{t-1})$, noise is injected into the motion model. The distance each particle moves on the surface is

$$u^i_t = u_t + \delta^i_u, \tag{7}$$

where δ^i_u is a random sample from $\mathcal{N}(0, \sigma^2_u)$. Also, the same method is used to advance the phase of each particle.

$$\phi^i_t = \phi^i_{t-1} + (\omega + \delta^i_\omega)dt, \tag{8}$$

where δ^i_ω is a random sample from $\mathcal{N}(0, \sigma^2_\omega)$.

3) Measurement Model: The measurement assumed in this work is the pose of the robot. In the case of a 2D map this pose is x and y Cartesian position, and a surface normal $\vec{n}$. The assumption is made that the angle measurement is the surface normal of the surface at the real location of the robot. The weight of each particle

$$w^i_t = p(z_t|s^i_t) = \left(\eta_z \exp \frac{dz^2}{\sigma^2_z}\right)\left(\eta_\theta \exp \frac{d\theta^2}{\sigma^2_\theta}\right), \tag{9}$$

where dz is the Cartesian distance between the current measurement z_t and the particles predicted measurement $\hat{z}^i$, and $d\theta$ angular distance between the measured surface normal and the particles predicted normal.

$$dz^2 = (z - \hat{z}^i)^T(z - \hat{z}^i) \tag{10}$$

$$d\theta = \arccos\left(\left|\vec{n} \cdot \vec{n}^i\right|\right) \tag{11}$$

4) Resampling: In order to decrease the risk of losing particle diversity, resampling only occurs when the variance of the particle weights is sufficiently large. Also, in order to reduce the sampling error, low variance sampling is used [9]. Instead of just drawing independent samples based on each particle's weight, this method ensures the survival of any particle which has a weight a weight greater than $\frac{1}{N}$, where N is the number of particles, is guaranteed to survive.

III. Experiments

In order to demonstrate the previously described framework, trials were run in the simulated system. The number of particles in each trial was set to N=1000, and the motion and observation models were as specified in section II-C. Motion model parameters were set to: $\sigma_u = 1$mm, and $\sigma_\omega = \frac{\pi}{180}$ radians. Observation model parameters were set to $\eta_z = 2$, and $\eta_\theta = 1$. Each trial was run for 360 iterations, equal to exactly one deformation period.

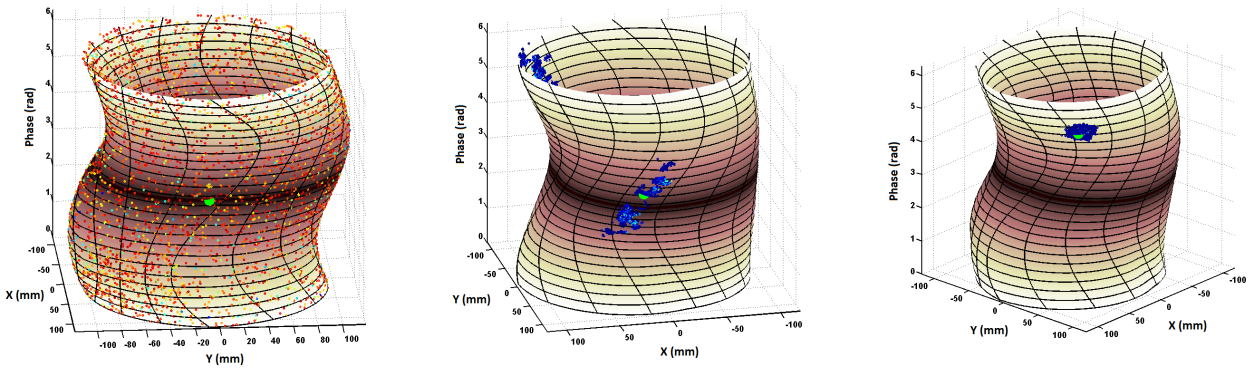
Fig. 3 illustrates the progression of the particle filter through a typical trial. The ground truth location of the robot is shown by the large green dot in each image. Fig. 3(a) shows the initialization of the filter, and demonstrates the randomized coverage of the state space. Fig 3(b) shows the particle filter after 25 iterations, where the particles have begun to form clusters around states with high likelihoods. The reason a few separate clusters form is due to geometric symmetries in the generated map, where different registration and phase parameters produce similar world-frame motion. Fig 3(c) shows the particle filter after 125 iterations, where enough observations have been gathered to greatly reduce the uncertainty, and only a single cluster remains.

To make a prediction for the map-frame location of the ground truth using the particle filter at any time during the trial, we take the 10 particles with the highest weights and remove the outliers, then average their state estimates. The reason we use an average of particles rather than simply outputting the single highest-weighted particle is for stability: as the weights on each particle is updated after every observation, the highest-weighted particle is likely to change often, and our localization output will be rather noisy.

State estimation errors made by the particle filter prediction output were recorded, and averaged over 100 trials to obtain a statistically significant result. Each of the 100 trials are completely independent with a different generated map, and separate initializations. The results are shown in Fig 4. We observe that on average the particle filter has converged by iteration 100, with almost all trials converging after 200 iterations. The average registration errors in position and angle are reduced to simply discretization errors in the map generation at 1.0mm and 0.02 rad, respectively. The map frame errors are slightly higher, with the average position error at 3mm and phase error at 0.05 rad.

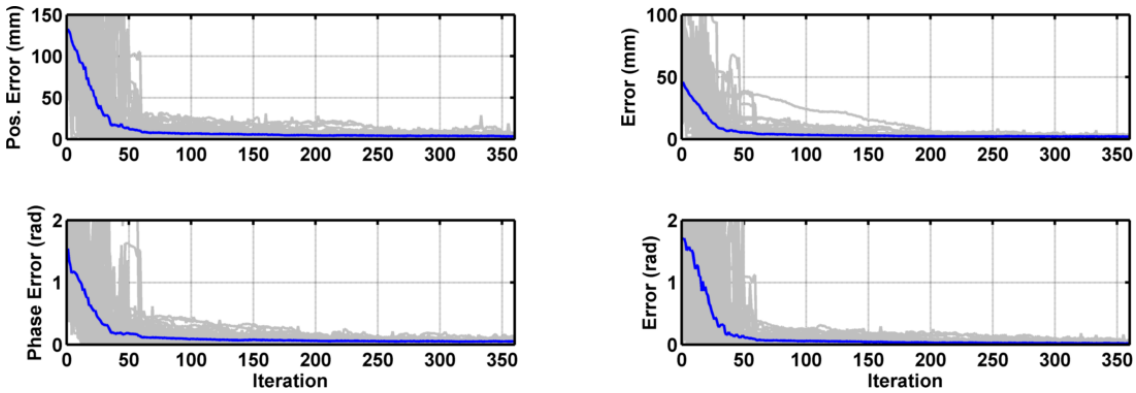
IV. Discussion

The presented work shows that estimation of registration and localization parameters on a periodically deforming surface is feasible, and that a well-tuned particle filter can



(a) (b) (c)

Fig. 3. Representative results of localizing on map using a particle filter. The ground truth position of the robot is shown by the large green dot. Each particle is represented by a small dot whose color denotes the particle weight. Low weights correspond to blue and high to red, with the color scale spanning the weights of the current particles. Plots correspond to (a) Filter initialization with particles randomly distributed over the surface, (b) after 25 iterations, and (c) after 125 iterations.



(a) (b)

Fig. 4. (a) Map frame error and phase error and (b) registration error for 2D localization over 100 runs. Gray lines denote a single trial, while blue lines denote the average over all trials. Errors for each run are calculated between the ground truth and the average state estimate of the 10 highest weighted particles. Position errors are calculated using the euclidean distance, phase and angle errors are absolute differences.

quickly and accurately converge to give low-error results. The estimation error decreases with higher-fidelity maps or finer discretization, and the steady-state error is close to the discretization error of the map itself. The maps used, however, are likely higher-fidelity than those derived from medical imaging technology, and future work will investigate using lower resolution maps or simultaneous localization and mapping. While this work was implemented in a simplified 2D example, the framework is extendable to 3D, and provides insight into the importance of shrinking the parameter space through intelligent particle initialization. Work implementing the presented framework on realistic 3D models of the beating heart is ongoing, as is incorporating deformation due to respiration. Future work is also planned to investigate how heartbeat irregularities affect performance.

REFERENCES

[1] M. Mack, "Minimally invasive cardiac surgery," *Surg. Endosc.*, vol. 20, pp. S488–S492, 2006.

[2] A. Thiagalingam, R. Manszke, A. D'Avila, I. Ho, A. H. Locke, J. N. Ruskin, R. C. Chan, and V. Y. Reddy, "Intraprocedural volume imaging of the left atrium and pulmonary veins with rotational x-ray angiography: Implications for catheter ablation of atrial fibrillation," *Journal of Cardiovascular Electrophysiology*, vol. 19, no. 3, pp. 293–300, 2008.

[3] R. A. Omary, J. D. Green, B. E. Schirf, Y. Li, J. P. Finn, and D. Li, "Real-time magnetic resonance imaging-guided coronary catheterization in swine," *Circulation*, vol. 107, no. 21, pp. 2656–2659, 2003.

[4] P. Novotny, J. Stoll, P. Dupont, and R. Howe, "Real-time visual servoing of a robot using three-dimensional ultrasound," in *IEEE Int. Conf. on Robotics and Automation*, Apr. 2007, pp. 2655 –2660.

[5] S. Tully, G. Kantor, and H. Choset, "Inequality constrained Kalman filtering for the localization and registration of a surgical robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Sep. 2011.

[6] A. Brij Koolwal, F. Barbagli, C. Carlson, and D. Liang, "An ultrasound-based localization algorithm for catheter ablation guidance in the left atrium," *Int. J. Robot. Res.*, vol. 29, no. 6, pp. 643–665, 2010.

[7] G. Shechter, J. Resar, and E. McVeigh, "Displacement and velocity of the coronary arteries: cardiac and respiratory motion," *IEEE Trans. Med. Imag.*, vol. 25, no. 3, pp. 369–375, 2006.

[8] N. Patronik, T. Ota, M. Zenati, and C. Riviere, "A miniature mobile robot for navigation and positioning on the beating heart," *IEEE Trans. Robot.*, vol. 25, no. 5, pp. 1109 –1124, Oct. 2009.

[9] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent robotics and autonomous agents. The MIT Press, Aug. 2005.