Planning Sensing Sequences for Subsurface 3D Tumor Mapping

Brian Y. Cho, Tucker Hermans, and Alan Kuntz

***Abstract*— Surgical automation has the potential to enable increased precision and reduce the per-patient workload of overburdened human surgeons. An effective automation system must be able to sense and map subsurface anatomy, such as tumors, efficiently and accurately. In this work, we present a method that plans a sequence of sensing actions to map the 3D geometry of subsurface tumors. We leverage a sequential Bayesian Hilbert map to create a 3D probabilistic occupancy model that represents the likelihood that any given point in the anatomy is occupied by a tumor, conditioned on sensor readings. We iteratively update the map, utilizing Bayesian optimization to determine sensing poses that explore unsensed regions of anatomy and exploit the knowledge gained by previous sensing actions. We demonstrate our method's efficiency and accuracy in three anatomical scenarios including a liver tumor scenario generated from a real patient's CT scan. The results show that our proposed method significantly outperforms comparison methods in terms of efficiency while detecting subsurface tumors with high accuracy.**

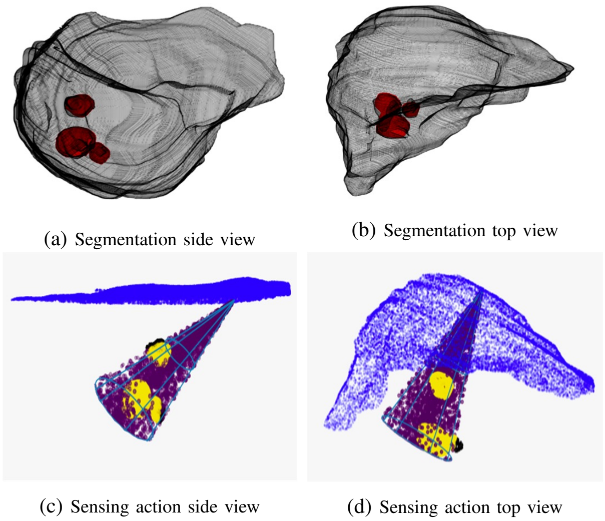
I. Introduction

Surgical automation [1], [2] has the potential to increase precision and reduce the per-patient workload of surgeons, a group already stretched thin by a general population that is rapidly outgrowing surgical resources [3], [4]. An effective autonomous surgical system must be able to sense, map, and reason about a patient's anatomy below the visible surface of organs.

Consider the case of resecting subsurface tumors in an organ. It is imperative to have an accurate understanding of the location and geometry of the tumors in the organ prior to resection in order to minimize the damage to healthy tissue while ensuring all cancerous tissue is safely removed (see Fig. 1). While pre-operative imaging techniques such as computed tomography (CT) can provide general knowledge of the anatomy, it may change before or during surgery. This necessitates the use of intraoperative sensing and mapping of the tumors, e.g., through the use of an ultrasound probe. Such sensing should be both accurate and efficient in order to reduce the overall time required for the surgical procedure. In this work, we present a method to enable an autonomous surgical system to accurately and efficiently map subsurface patient anatomy, such as tumors inside an organ.

Specifically, we present a method that utilizes a probabilistic model of anatomical geometry to iteratively determine sensing actions that improve the model's understanding of the geometry. To do so, we wrap a Bayesian optimization [5] framework around a probabilistic representation of 3D geometry, called a Bayesian Hilbert map [6], [7], that utilizes sensor information to determine the likelihood that any particular point in the organ is occupied by a tumor. Bayesian Hilbert maps enable us to iteratively update the occupancy probability of a given continuous anatomical environment as we obtain sensor readings. The Bayesian Hilbert map then serves as a continuous posterior distribution on tumor occupancy at each iteration, enabling the Bayesian optimization to determine the next sensing action. Our method utilizes an objective function in the optimization that balances the exploration of unsensed regions of anatomy while reducing the uncertainty around regions where tumors have already been identified. In this way the method is able to accurately map the geometry of the subsurface tumors in a relatively few number of sensing steps.

While prior work has focused on sensing and mapping in 2D [8], [9], our use of Bayesian Hilbert maps enables reasoning over 3D geometry accurately and efficiently. We evaluate the performance of our method in multiple surgery-inspired scenarios, including environments in which tumors are randomly placed, evaluating generalizability, as well as a real-life scenario generated via the liver tumor segmentation (LiTS) dataset [10]. We compare our method to random sampling and a multi-resolution scan. Results show that our proposed method outperforms both comparison methods, significantly reducing the number of sensing actions required for accurately mapping the embedded tumors.


(a) Segmentation side view

(b) Segmentation top view

(c) Sensing action side view

(d) Sensing action top view

Fig. 1: Example tumor localization scenario. (a,b) A liver (black) and 3 tumors (red) segmented from a CT scan in the liver tumor segmentation (LiTS) dataset. (c,d) A cone-shaped sensing volume (purple) normal to the liver surface (blue) detects the embedded tumors (yellow).

The authors are with the Robotics Center and School of Computing, University of Utah, Salt Lake City, UT 84112, USA. TH is also affiliated with NVIDIA. {brian.cho,tucker.hermans,alan.kuntz}@utah.edu

II. Related Work and Background

A variety of methods have been developed to automate several tasks in robotic surgery. Jansen *et al.* [11] developed an automated method of tissue retraction to grasp deformable objects using a spring model. Elek *et al.* [12] automated blunt dissection of tissue using motion primitives. McKinley *et al.* [13] developed an interchangeable surgical tool system to automate a multi-step tumor resection including palpation, incision, debridement, and injection. In this paper, we focus on autonomous sensing for tumor localization.

Automated sensing for tumor localization and segmentation has also been studied via palpation. Garg *et al.* [8] presented an algorithm that samples over a stiffness map for localizing tumor boundaries with a palpation probe. Nichols *et al.* [9] automated robotic palpation to localize and segment hard regions in soft tissues, also for tumor localization. These methods are specific to palpation, framing the problem in two dimensions. In this work, we consider a 3D sensor, such as a swept ultrasound, and model the geometry in a 3D continuous space.

Global optimization can enable autonomous surgical systems to optimize throughout a given search space by finding the global extremum of a given function. Bayesian optimization [5] is one popular global optimization method and has been widely used in applications such as object surface estimation [14] and hyper-parameter optimization [15]. Bayesian optimization has also been used in surgical robotics, for instance as the optimization method in the above mentioned palpation work of Garg *et al.* [8]. Particularly effective in cases where function evaluation is computationally expensive, Bayesian optimization decides which points in the search space should be sampled and evaluated next via acquisition functions, such as expected improvement (EI) [16]. In this work, we leverage Bayesian optimization to determine our sensing poses, choosing an acquisition function that considers the balance between exploration and exploitation. Since autonomous surgical systems may have limited or inaccurate prior information regarding the number and size of tumors present in a given anatomical environment, it is necessary to simultaneously explore areas of high uncertainty as well as exploit existing knowledge of areas where tumors have already been sensed.

A variety of methods have been used to model a probabilistic distribution over a robot's environment. For instance, Gaussian processes [17] have been used with Bayesian optimization as a standard modeling method. Garg *et al.* [8], mentioned above, build a probabilistic model of the tissue stiffness map using Gaussian processes. There have been other methods, e.g., [18], which model the probabilistic occupancy state of an environment using Gaussian Processes. Senanayake *et al.* [7] developed Bayesian Hilbert maps to build an occupancy map in dynamic environments. They simultaneously introduced an extended version of Bayesian Hilbert maps, sequential Bayesian Hilbert maps [7], as a fast, sequential long-term occupancy mapping method in dynamic environments. In this work, we leverage sequential Bayesian Hilbert maps to build a probabilistic occupancy map updated sequentially via sensing.

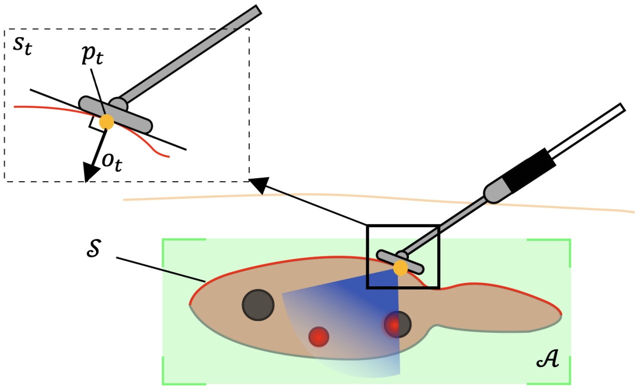


Fig. 2: Example of the sensor pose at iteration t. The sensing pose $\mathbf{s}_t$ is composed of position $\mathbf{p}_t$ (yellow dot) and orientation $\mathbf{o}_t$ (black arrow)

III. Problem Formulation

In this work we assume that the sensor is noiseless, that segmentation in a sensed volume is perfect, and that the time required to perform the sensing action and associated segmentation dominates the time required to move the sensor between sensing poses. We also assume that anatomy is rigid such that there is no deformation during sensing.

We consider a case where tumors are embedded in an anatomical environment $\mathcal{A} \subset \mathbb{R}^3$, e.g., a patient's organ. We define the tumors as $\mathcal{T} \subset \mathcal{A}$, a possibly disconnected set of arbitrary geometry. We define a sensor workspace $\mathcal{S}$ that includes N possible sensing poses $\mathbf{s}_i$ for $i = 1, \cdots, N$. A sensing pose $\mathbf{s}_i$ is a vector concatenating position and orientation, i.e., $\mathbf{s}_i = [\mathbf{p}_i, \mathbf{o}_i]$, where $\mathbf{p}_i \in \mathbb{R}^3$ and $\mathbf{o}_i \in SO(3)$ are the sensing position and orientation, respectively (see Fig. 2). We define a general sensor model as the set of points, $v_i \subset \mathbb{R}^3$ that are sensed during a sensing action performed at $\mathbf{s}_i$, e.g., the volume visualized by an ultrasound sensing action performed when the transducer is centered and oriented at $\mathbf{s}_i$.

Let $\mathcal{S}_\mathcal{M} \subset \mathcal{S}$ then be an ordered sequence of k sensing actions where $\mathcal{S}_\mathcal{M} = \{\mathbf{s}_1, \cdots, \mathbf{s}_t, \cdots, \mathbf{s}_k\}$ and $|\mathcal{S}_\mathcal{M}| = k \leq N$. The sensed volume of a given sequence $\mathcal{S}_\mathcal{M}$ is then

$$\mathcal{V}_{\mathcal{S}_\mathcal{M}} = \bigcup_{i=1}^{k} v_i.$$

The goal then is to determine a sequence of sensing actions $\mathcal{S}_\mathcal{M}$ of minimal length (i.e., $|\mathcal{S}_\mathcal{M}|$), such that the geometry of all the tumors are mapped with high certainty.

IV. Method

At a high level, our method is composed of two main pieces: (i) a 3D probabilistic occupancy map, implemented as a posterior likelihood distribution representing the likelihood that any point in space is occupied by a tumor, and (ii) an iterative optimization-based framework that reasons over the current distribution, determines the next sensing location, performs the sensing, and updates the distribution

```
Algorithm 1: Sensor Sequence Planning
Input: Sensor workspace $\mathcal{S}$, Search space $\mathcal{A}$
Output: A sequence of sensing configurations $\mathcal{S}_{\mathcal{M}}$
initialize $\omega \leftarrow \mu_0, \Sigma_0$
$t \leftarrow 0$
while time remains do
    $t \leftarrow t + 1$
    if $t$ = 1 then
        $\mathbf{s}_t \leftarrow$ random($\mathcal{S}$)
        $\mathcal{S}_{\mathcal{M}} \leftarrow \mathbf{s}_t$
    else
        $\mathbf{s}_t \leftarrow$ NextQuery($t$, $\omega$, $\mathcal{S}$, $\mathbf{x}_*$)
        $\mathcal{S}_{\mathcal{M}} \leftarrow$ concatenate($\mathcal{S}_{\mathcal{M}}, \mathbf{s}_t$)
    end
    Acquire sensor data $\mathcal{D}_t$ given $\mathbf{s}_t$
    $\mu_t, \Sigma_t \leftarrow$ learn_parameters($\mathcal{D}_t$, $\omega$)
    $\omega \leftarrow \mu_t, \Sigma_t$
end
return $\mathcal{S}_{\mathcal{M}}$
```

```
Algorithm 2: NextQuery
Input: $t$, $\omega$, $\mathcal{S}$, $\mathbf{x}_*$
Output: Next sensing pose $\mathbf{s}_t$
$\mathbf{a}_t \leftarrow \arg\max_{\mathbf{x}_*} EI(\mathbf{x}_*, \omega)$
$\mathbf{s}_t \leftarrow$ next_sensing_pose($a_t$, $\mathcal{S}$)
return $\mathbf{s}_t$
```

based on what was sensed. Our method performs (ii) in a loop, iteratively updating (i) to improve the occupancy map, localizing the tumor(s) quickly and accurately. The method is outlined in Algorithm 1.

A. Sensor Model

In this work we consider a sensing action at sensing pose $\mathbf{s}_i \in \mathcal{S}$ to be a cone-shaped volumetric occupancy map of the anatomy in the cone. In practice this could come from automated segmentation of a localized ultrasound sensing action, for instance. This cone then defines the sensor volume v_i (see Fig. 3). Points within the cone are then labeled as either occupied (i.e., part of tumor geometry), or unoccupied (i.e., not tumor). More formally, at iteration t, for each point in the cone $\mathbf{x} \in v_t$ we define an occupancy indicator $y \in \{0, 1\}$ denoting whether $\mathbf{x}$ is sensed as part of a tumor. This then defines the tuple $(\mathbf{x}, y)$ for each point in v_t. We define the set of these tuples for a given sensing action as the sensor data $\mathcal{D}_t$. The sensor data $\mathcal{D}_t$ acquired from each measurement is used to update the sequential Bayesian Hilbert map parameters which define a probabilistic occupancy map of the unsensed anatomy, described below.

B. 3D Occupancy Mapping

We leverage sequential Bayesian Hilbert maps to model the occupancy states of given anatomical environments in an iterative manner. Essentially, sequential Bayesian Hilbert maps define a classifier that estimates the probability of an unsensed point $\mathbf{x}$ being occupied. Kernel functions define the sequential Bayesian Hilbert map features where for some kernel k, $k(\mathbf{x}, \widetilde{\mathbf{x}}_j)$, the kernel evaluates the similarity between the query point $\mathbf{x}$ and a hinge point $\widetilde{\mathbf{x}}_j$ that is fixed at some location in the search space to be mapped. The feature vector $\Psi(\mathbf{x})$ represents the vector of kernel evaluations to all hinge points in the space, $\Psi(\mathbf{x}) = (k(\mathbf{x}, \widetilde{\mathbf{x}}_1), k(\mathbf{x}, \widetilde{\mathbf{x}}_2), \cdots)$. Following [7], we fix M hinge points $\widetilde{\mathbf{x}}_j$ for $j = 1, \cdots, M$ spatially in $\mathcal{A}$ to compute the feature vector $\Psi(\mathbf{x}) \in \mathbb{R}^{1\times M}$. We then define the likelihood of occupancy using the feature vector via a parametric logistic-regression model,

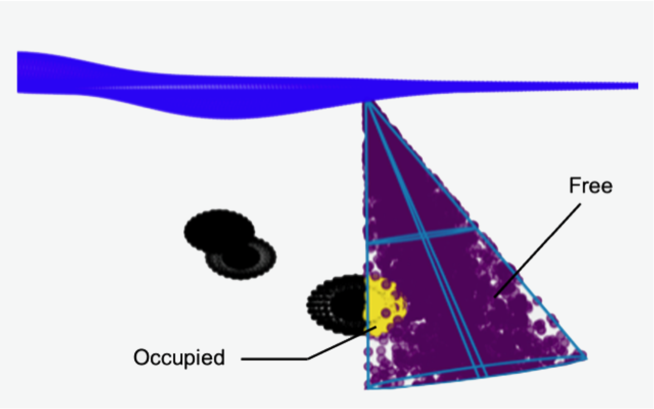


Fig. 3: Example sensing action. In the cone-shaped sensing volume, oriented normal to the surface, the unoccupied free points (purple) and occupied tumor points (yellow) are determined.

$$P(y|\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{w}\Psi^T(\mathbf{x})),$$

where $\sigma(\cdot)$ is the sigmoid function and $\mathbf{w} \in \mathbb{R}^{1\times M}$ is a linear weight vector. After sensing observation t of data collection, we model a normal distribution over the weight vector, $\mathbf{w} \sim \mathcal{N}(\mu_t, \Sigma_t)$, with mean $\mu_t \in \mathbb{R}^{1\times M}$ and variance $\Sigma_t \in \mathbb{R}^{1\times M}$. We define the vector of all parameters $\omega = \{\mu, \Sigma\}$. Note that in sequential Bayesian Hilbert maps the weight vector $\mathbf{w}$ is typically initialized with zero mean and high variance, representing that we do not have prior knowledge on the distribution. We use the squared exponential kernel in our implementation

$$k(\mathbf{x}, \widetilde{\mathbf{x}}) = \exp\left(-\gamma||\mathbf{x} - \widetilde{\mathbf{x}}||^2\right) \tag{1}$$

where γ is a hyper-parameter defining the length scale of the kernel.

Given the model we then wish to learn the mean and variance of the parameter distribution ω. As its name suggests, sequential Bayesian Hilbert maps use Bayes' theorem to model the posterior distribution of the parameter ω as:

$$P(\mathbf{w}|\mathbf{x}, y) = \frac{P(y|\mathbf{x}, \mathbf{w})P(\mathbf{w})}{P(y)}.$$

The posterior over weights $P(\mathbf{w}|\mathbf{x}, y)$ cannot be explicitly computed because of the combination of the sigmoidal likelihood and Gaussian prior, [7] provides a way to approximate the posterior $Q(\omega)$ defined also as Gaussians by estimating the parameters ω through expectation-maximization (EM).

The `learn_parameter(·)` function in Algorithm 1 is a function for estimating the parameters ω. (See [7] for further details).

This then defines a continuous probabilistic occupancy map, e.g., for any query point $\mathbf{x}_* \in \mathcal{A}$, the probability of occupancy is defined as $P(y|\mathbf{x}_*, \mathbf{w})$.

C. Iterative Optimization-Based Framework

Given the sequential Bayesian Hilbert map at a given iteration, we must determine the next sensing pose to gather the most relevant information and update the map. This process is outlined in Algorithm 2. The sequential Bayesian Hilbert map provides a probabilistic occupancy map defined over the 3D anatomy $\mathcal{A}$, but the sequential Bayesian Hilbert map has no knowledge of our sensor workspace $\mathcal{S}$. As such, we will first determine a point in $\mathcal{A}$ from the sequential Bayesian Hilbert map that should be sensed next, and then determine a pose in $\mathcal{S}$ that will do so effectively.

We leverage Bayesian optimization to determine the next query point $\mathbf{a}_t \in \mathcal{A}$. To do so, we must define an acquisition function based on the posterior distribution provided by the sequential Bayesian Hilbert map at the current iteration. As is frequently the case in Bayesian optimization, in our problem it is important to define an acquisition function that balances exploitation and exploration. Insufficient exploitation may lead to failure to fully map the tumors while insufficient exploration may result in not mapping tumors in unexplored regions. To balance these, we choose Expected Improvement (EI) [16] as our acquisition function.

More formally, EI is defined as

$$EI(\mathbf{x}_*) = \mathbb{E}[\max(0, f(\mathbf{x}_*) - f(\mathbf{x}^+))],$$

where $f(\cdot)$ is an objective function which in our method is the occupancy likelihood distribution, $P(y|\mathbf{x}_*, \mathbf{w})$, defined by the Bayesian Hilbert map at the current iteration, and $f(\mathbf{x}^+)$ is the highest value of the distribution. The expected improvement can be expressed in closed form [16]:

$$\begin{aligned} EI(\mathbf{x}_*, \omega) = (\mu - f(\mathbf{x}^+) - \xi)\Phi(\frac{\mu - f(\mathbf{x}^+) - \xi}{\sigma}) \\ + \Sigma\phi(\frac{\mu - f(\mathbf{x}^+) - \xi}{\sigma}) \end{aligned} \quad (2)$$

where $\Phi(\cdot)$ is the cumulative distribution function, $\phi(\cdot)$ is the probability density function, and ξ is an exploration parameter. Mean μ and standard deviation Σ are also computed via the sequential Bayesian Hilbert map. We are able to determine the next query point $\mathbf{a}_t \in \mathcal{A}$ by optimizing the acquisition function, $\mathbf{a}_t = \arg\max_{\mathbf{x}_*} EI(\mathbf{x}_*, \omega)$.

We next must determine the sensing pose $\mathbf{s}_t$ corresponding to the query point $\mathbf{a}_t$ (`next_sensing_pose` in Algorithm 2, line 2). To do so, we choose a sensing pose that has an orientation that closely aligns with the vector defined by the pose's position and the query point, i.e., a pose that points toward the query point (see Fig. 4). We search over the sensing poses and for each define a query vector $\mathbf{q}$ as the vector between the given sensing pose's position $\mathbf{p}$ and the query point $\mathbf{a}_t$. We then determine the angular difference between the sensing pose's orientation $\mathbf{o}$ and $\mathbf{q}$. We select the first sensing pose found such that this angle is below a given threshold.

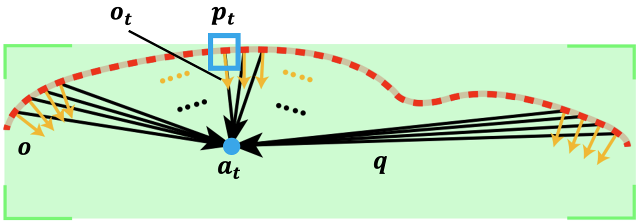


Fig. 4: Determining the next sensing pose $\mathbf{s}_t = (\mathbf{p}_t, \mathbf{o}_t)$. When a query vector (black vectors) and a sensing pose's surface normal (orange vectors) are aligned (blue box), the corresponding pose is selected.

D. Combined Method

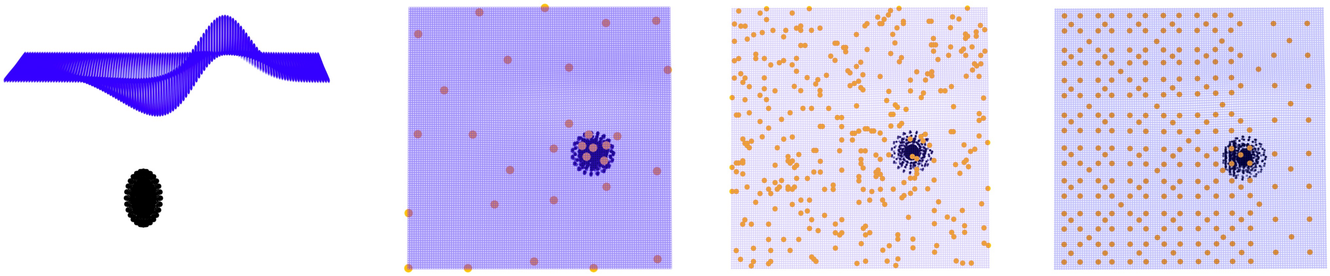
Combining the pieces above we get the full method, outlined in Algorithm 1 and Algorithm 2. The method takes as input the sensor workspace $\mathcal{S}$ and the region to search over $\mathcal{A}$, and outputs a sequence of sensing configurations $\mathcal{S}_{\mathcal{M}}$ to detect the tumor(s) $\mathcal{T}$ embedded in the anatomical environment. We first initialize the parameter ω of each kernel of the Bayesian Hilbert maps (Algorithm 1 line 1). The first sensing pose $\mathbf{s}_1$ is then randomly chosen from $\mathcal{S}$ as we do not yet have knowledge of the distribution (Algorithm 1 line 6). Afterwards, sensing poses are determined by Algorithm 2 (called by line 9 of Algorithm 1) and the sequence of poses is accumulated in $\mathcal{S}_{\mathcal{M}}$ (`concatenate` in Algorithm 1, line 10). For each sensing pose, the corresponding sensor data $\mathcal{D}_t$ is collected (Algorithm 1 line 12) and is then used to update the posterior distribution of the parameter ω (Algorithm 1 line 13). The method repeats as time allows, refining the estimation of the tumor volumes with increasing iterations.

V. Experiments and Results

We evaluate our method in two ways. First, we evaluate the method's efficiency and compare against random sampling and a multi-resolution scan as strategies for determining sensing pose sequences. We do so in synthetic, randomly generated example scenarios and demonstrate our method is capable of localizing the embedded tumors with significantly fewer sensing actions. Second, we evaluate our method's accuracy as a function of the number of sensing actions in the presence of multiple tumors. We do so both in a synthetic environment with randomly placed tumors and in a real medical scenario segmented from a patient CT scan in the liver tumor segmentation (LiTS) dataset [10]. For all experiments we set the parameters $\gamma = 5$ in the kernel function (1), $\xi = 0.01$ in the acquisition function (2), and use 2890 hinge points distributed on a 3D grid in the environments.

A. Evaluating Efficiency

To evaluate our method's sampling efficiency quantitatively, we generate a synthetic organ surface and place a spherical tumor volume randomly below it (see Fig. 5 (a)). We refer to this as scenario 1. The randomly placed



(a) Example of scenario 1

(b) Our method

(c) Random sampling

(d) Multi-resolution scan

Fig. 5: Scenario 1, trial 1. The dots (orange) indicate the sensing poses chosen by the methods to identify the embedded tumor. (a) The surface (blue) and tumor (black). (b) Our method requires 31 samples while balancing the exploration of the entire search space and the exploitation of regions where tumors may lie. (c) Random sampling requires 558 samples. (d) Multi-resolution scan requires 270 samples.

sphere-shaped tumor is composed of 500 data points and the sensor workspace is composed of 14,400 data points. The synthetic surface is made uneven to approximate the uneven nature of the surface of human organs and is generated via the `get_test_data` function provided by Matplotlib, a graphical plotting library for Python. We define the positions in $\mathcal{S}$ as the points on the surface and compute the orientation of each sensing pose as the surface normal perpendicular to a tangent plane fit to each surface point's 10 nearest neighbors.

We compare our method against random sampling, in which we draw sample poses uniformly at random from $\mathcal{S}$; and a coarse-to-fine multi-resolution scan, in which the search space is divided into increasingly fine cells in each round, i.e., in the first round there is one cell containing the entire search space, in the second round the space is divided into four cells, etc. At each round we apply sensing actions to the center of each cell, ordered as a raster scan. We measure the number of samples required by each method to detect the tumor, defined as having sensed 95% of the tumor points.

We average the results over ten trials in which the tumor is randomly placed below the surface. We demonstrate the results for all methods across all trials in Fig. 6. As can be seen, our method detects the tumor with much fewer samples. Compared to our method, the random sampling method required on average approximately 10.1 times the number of sensing actions (258.6 required by random sampling compared with 25.5 required by our method) and the multi-resolution scan required approximately 6.5 times the number of sensing actions (172.2 required by the multi-resolution scan compared with 25.5 required by our method) to fully sense the tumor. The specific sensing poses determined by the method for one of the trials are shown in Fig. 5.

This analysis demonstrates our method's ability to efficiently generate sensor poses that sense the tumor geometry.

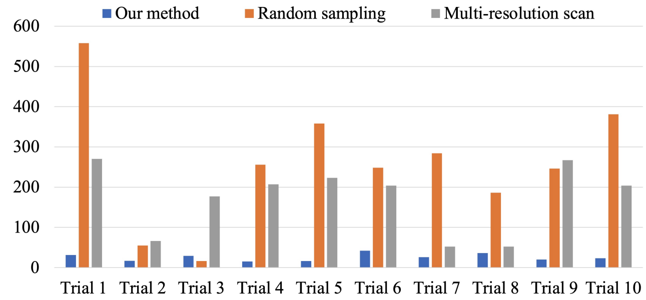


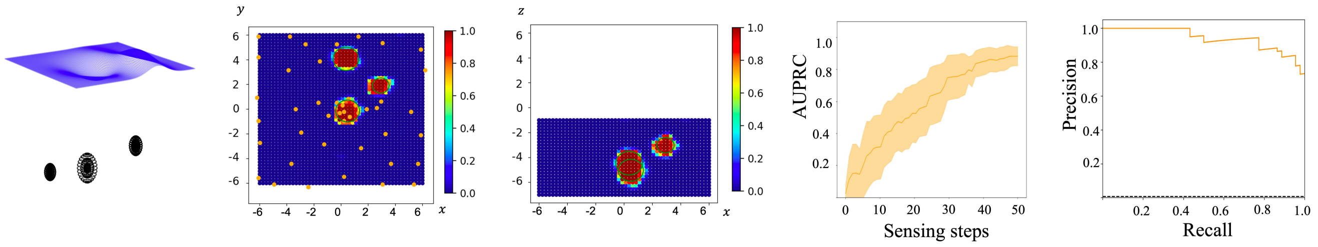
Fig. 6: The number of sensing pose samples required by each method for 10 random trials. The mean and standard deviation across the trials are 25.5 ± 9.0 for our method, 258.8 ± 156.7 for random sampling, and 172.2 ± 84.7 for the multi-resolution scan.

B. Evaluating Accuracy

A notable property of the Bayesian Hilbert occupancy map is that it incorporates probabilistic information about regions not yet sensed, which is refined with more sensing actions. Here we evaluate the accuracy of the occupancy map generated by our method as the number of sensing actions performed increases. We consider two scenarios, a synthetic scenario with three tumors randomly placed, named scenario 2 (see Fig 7 (a)), and a real-life scenario with multiple tumors in a patient's liver, segmented via 3D Slicer [19] from a CT scan in the liver tumor segmentation (LiTS) dataset [10], which we name the LiTS scenario (see Fig. 8 (a) and Fig. 9 (b)). $\mathcal{S}$ for scenario 2 is generated as for scenario 1, and $\mathcal{S}$ for the LiTS scenario is generated by taking points on the top surface of the liver segmentation (see Fig. 8 (a)) and defining surface normals as in scenarios 1 and 2.

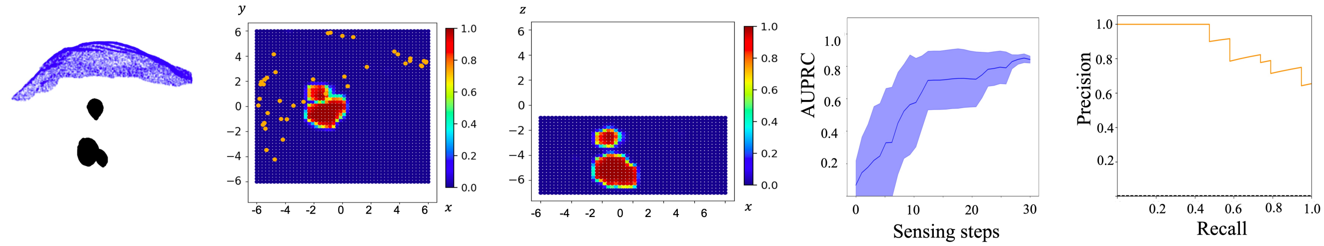
We note that the probabilistic occupancy map being produced by our method acts as a binary classifier when combined with a threshold value. As a metric to evaluate binary classifiers, both the area under the receiver operating characteristics (AUROC) curve and the area under the precision-recall curve (AUPRC) are popular metrics. However, the ROC tends to provide an optimistic view of the performance when it comes to imbalanced datasets, potentially resulting in incorrect interpretation [20]. As our case is highly imbalanced, e.g., approximately a 200:1 ratio of free space to tumor volume in scenario 2 and a 300:1 ratio in the LiTS scenario, we choose the AUPRC as our evaluation metric.

For both scenario 2 and the LiTS scenario, we evaluate the AUPRC at each iteration, i.e., sensing action, and evaluate how the AUPRC improves as the iterations increase. At each iteration the Bayesian Hilbert occupancy map is refined, becoming more accurate, and the AUPRC improves. As our method is subject to random initialization, we average the



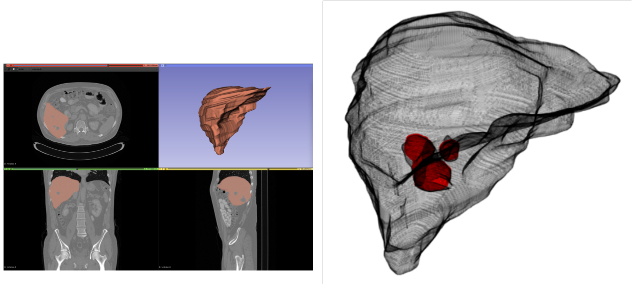
(a) Example of scenario 2 (b) Projection onto x-y plane (c) Projection onto x-z plane (d) AUPRC for 10 trials (e) Precision-recall curve at iteration 50 of the 5th trial

Fig. 7: (a) Scenario 2 consists of three sphere-shaped tumors and the same sensor workspace as scenario 1. The tumors are randomly positioned in each trial. (b) Top view (x-y plane) of the final occupancy map for the 5^{th} trial. The orange dots indicate the sensing poses chosen by our method. (c) The side view (x-z plane) of the occupancy map. (d) The mean (orange line) and standard deviation (shaded area) of the AUPRC as the iterations increase, averaged across 10 trials. The AUPRC converges to 0.88 within 50 sensing iterations for all trials. (e) The precision-recall curve at the last iteration of the 5^{th} trial, with AUPRC of 0.89.



(a) LiTS scene (b) Projection onto x-y plane (c) Projection onto x-z plane (d) AUPRC for 10 trials (e) Precision-recall curve at iteration 30 of the 5th trial

Fig. 8: (a) The LiTS scenario is composed of three tumors in the liver. (b) Top view of the final occupancy map of the 5^{th} trial. The orange dots indicate the sensing poses chosen by our method. (c) The side view of the occupancy map. (d) The mean (blue line) and standard deviation (shaded area) of the AUPRC as the iterations increase, averaged across 10 trials. The AUPRC graph converges to 0.84 within 30 steps for all trials. (e) The precision-recall curve at the last iteration of the 5^{th} trial, with AUPRC of 0.84.



(a) CT scan and segmentation (b) 3D rendering

Fig. 9: Liver and tumors segmented from the LiTS data. (a) The liver and tumors were segmented via 3D Slicer [19]. (b) The resulting 3D geometry utilized for evaluation.

results across 10 random trials with different initializations.

Fig. 7 shows the results for scenario 2. To evaluate correct classification we discretize the search space into 6292 points on a 3D grid. Fig. 7 (d) shows the mean and standard deviation of the AUPRC for 10 trials as the sensing iterations increase, converging to 0.88 at 50 iterations. A high AUPRC indicates that the method correctly labels the positive examples (tumors) without falsely labeling the negative examples (free space) as positive. Considering that we have a highly imbalanced environment, the baseline of an uninformed model would be 0.006 (the ratio of positives to negatives). The resulting AUPRC of 0.88 indicates high accuracy. Fig. 7 (e) shows the specific precision-recall curve at iteration 50 for one of the trials, and Fig. 7 (b) and (c) show two views of the occupancy map.

Fig. 8 shows the results for the LiTS scenario. We task the method with correctly classifying the tumors inside the liver (using the tumor segmentations as ground truth). In this scenario we discretize the liver search space into 6292 evenly spaced points on a 3D grid for evaluation. As with scenario 2, we average over 10 runs. In the LiTS scenario our method converges to an average of 0.84 AUPRC after 30 sensing iterations, demonstrating high accuracy when compared with the baseline of 0.003 of an uninformed model for this scenario.

VI. Conclusion

In this work, we presented a method for planning sensing actions that models the anatomical environment using sequential Bayesian Hilbert maps and determines the sensing strategy using Bayesian optimization. We evaluated our method in three anatomical scenarios, a synthetic scenario with a single tumor, a synthetic scenario with multiple tumors, and a real-life case of tumors in a patient's liver segmented from a CT scan. In the first scenario, we evaluated the efficiency of our method by comparing it with random sampling and a multi-resolution scan, showing that it outperforms other methods with respect to the number of sensing actions required. In the other two scenarios, we demonstrated the ability of our method to accurately map the tumors.

In future work, we plan to relax our assumptions. This includes augmenting our method with the ability to consider deformable organs and other anatomical features beyond tumors. We also plan to integrate real sensing via, e.g., ultrasound and implement the method on a physical robot. Further, we plan to extend the method to consider the sensing path rather than just discrete sensing actions.

References

[1] G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, and R. Merrifield, "The grand challenges of Science Robotics," *Science Robotics*, vol. 3, no. 14, p. eaar7650, 2018.

[2] M. Yip and N. Das, "Robot autonomy for surgery," in *The Encyclopedia of Medical Robotics*, Aug. 2018, pp. 281–313.

[3] S. Moffatt-Bruce, J. Crestanello, D. P. Way, and T. E. Williams Jr, "Providing cardiothoracic services in 2035: Signs of trouble ahead," *The Journal of Thoracic and Cardiovascular Surgery*, vol. 155, no. 2, pp. 824–829, 2018.

[4] E. C. Ellison, B. Satiani, D. P. Way, W. M. Oslock, H. Santry, and T. E. Williams, "The continued urbanization of American surgery: A threat to rural hospitals," *Surgery*, Aug. 2020.

[5] M. Pelikan, D. E. Goldberg, and E. Cantú-Paz, "BOA: The Bayesian optimization algorithm," in *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99*, vol. 1. Citeseer, 1999, pp. 525–532.

[6] F. Ramos and L. Ott, "Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1717–1730, 2016.

[7] R. Senanayake and F. Ramos, "Bayesian hilbert maps for dynamic continuous occupancy mapping," in *Conference on Robot Learning*. PMLR, 2017, pp. 458–471.

[8] A. Garg, S. Sen, R. Kapadia, Y. Jen, S. McKinley, L. Miller, and K. Goldberg, "Tumor localization using automated palpation with gaussian process adaptive sampling," in *IEEE International Conference on Automation Science and Engineering (CASE)*, 2016, pp. 194–200.

[9] K. A. Nichols and A. M. Okamura, "Methods to segment hard inclusions in soft tissue during autonomous robotic palpation," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 344–354, 2015.

[10] P. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen, Q. Dou, C.-W. Fu, X. Han, P.-A. Heng, J. Hesser, S. Kadoury, T. Konopczynski, M. Le, C. Li, X. Li, J. Lipkovà, J. Lowengrub, H. Meine, J. H. Moltz, C. Pal, M. Piraud, X. Qi, J. Qi, M. Rempfler, K. Roth, A. Schenk, A. Sekuboyina, E. Vorontsov, P. Zhou, C. Hülsemeyer, M. Beetz, F. Ettlinger, F. Gruen, G. Kaissis, F. Lohöfer, R. Braren, J. Holch, F. Hofmann, W. Sommer, V. Heinemann, C. Jacobs, G. E. H. Mamani, B. van Ginneken, G. Chartrand, A. Tang, M. Drozdzal, A. Ben-Cohen, E. Klang, M. M. Amitai, E. Konen, H. Greenspan, J. Moreau, A. Hostettler, L. Soler, R. Vivanti, A. Szeskin, N. Lev-Cohain, J. Sosna, L. Joskowicz, and B. H. Menze, "The liver tumor segmentation benchmark (LiTS)," 2019.

[11] R. Jansen, K. Hauser, N. Chentanez, F. Van Der Stappen, and K. Goldberg, "Surgical retraction of non-uniform deformable layers of tissue: 2d robot grasping and path planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 4092–4097.

[12] R. Elek, T. D. Nagy, D. Á. Nagy, T. Garamvölgyi, B. Takács, P. Galambos, J. K. Tar, I. J. Rudas, and T. Haidegger, "Towards surgical subtask automation—blunt dissection," in *IEEE 21st International Conference on Intelligent Engineering Systems (INES)*, 2017, pp. 000 253–000 258.

[13] S. McKinley, A. Garg, S. Sen, D. V. Gealy, J. P. McKinley, Y. Jen, M. Guo, D. Boyd, and K. Goldberg, "An interchangeable surgical instrument system with application to supervised automation of multilateral tumor resection," in *IEEE International Conference on Automation Science and Engineering (CASE)*, 2016, pp. 821–826.

[14] Z. Yi, R. Calandra, F. Veiga, H. van Hoof, T. Hermans, Y. Zhang, and J. Peters, "Active tactile object exploration with gaussian processes," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4925–4930.

[15] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *25th Annual Conference on Neural Information Processing Systems (NIPS)*, vol. 24. Neural Information Processing Systems Foundation, 2011.

[16] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, 1998.

[17] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Springer, 2003, pp. 63–71.

[18] S. T. O'Callaghan and F. T. Ramos, "Gaussian process occupancy maps," *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 42–62, 2012.

[19] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka, *et al.*, "3D Slicer as an image computing platform for the Quantitative Imaging Network," *Magnetic Resonance Imaging*, vol. 30, no. 9, pp. 1323–1341, 2012.

[20] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PloS one*, vol. 10, no. 3, p. e0118432, 2015.