2025 IEEE International Conference on Robotics and Automation (ICRA 2025)
May 19-23, 2025. Atlanta, USA

Heterogeneous Sensor Fusion and Active Perception for Transparent Object Reconstruction with a PDM^2 Sensor and a Camera

Fengzhi Guo, Shuangyu Xie, Di Wang, Cheng Fang, Jun Zou, and Dezhen Song

Abstract— **Transparent household objects present a challenge for domestic service robots, since neither regular cameras nor RGB-D cameras can provide accurate points for shape reconstruction. The new type of pretouch dual-modality distance and material sensor (PDM^2) can provide reliable and accurate depth readings, but it is a point sensor and scanning the object exclusively with the sensor is too inefficient. Hence, we present a sensor fusion approach by combining a regular camera with the PDM^2 sensor. The approach is based on a data fusion algorithm for shape reconstruction and an active perception algorithm for scan planning for the PDM^2 sensor. The data fusion algorithm is a distributed Gaussian process (GP)-based shape reconstruction method that allows for incremental local update to reduce computational time. The active perception algorithm is an optimization-based approach by increasing the information gain (IG) and prioritizing the boundary points under a preset travel distance constraint. We have implemented and tested the algorithms with six different transparent household items. The results show satisfactory shape reconstruction results in all test cases with an average increase in intersection over union (IoU) from 0.73 to 0.96.**

I. Introduction

Robust handling of household objects is a fundamental capability in domestic robotic applications. The ubiquitous presence of transparent objects in a common household, such as glass cups, plastic bottles, etc., challenges existing sensing modalities, such as a camera, due to strong refraction and reflection in the light path. Therefore, the object shape reconstructed from camera images often contains significant errors (see Fig. 1(a)). Recently, we have developed a new type of pretouch dual-modality distance and material sensor (PDM^2) [1]–[3] to deal with transparent objects. The new sensor utilizes pulse echo ultrasound (US) and optoacoustic (OA) modalities to improve its capabilities and achieve submillimeter-level accuracy in ranging. However, the PDM^2 sensor is a point sensor and scanning the shape of the object would be too slow for grasping applications.

An immediate thought is to develop a sensor fusion approach that combines a camera with the PDM^2 sensor that balances both accuracy and speed for object shape reconstruction. This requires us to address two issues. The first issue is to develop a data fusion algorithm that fuses a large number of noisy points from image-based reconstruction with a small number of accurate points from the PDM^2 sensor. We present a distributed Gaussian process (GP)-based shape reconstruction method that allows incremental local update to reduce computation time. The second issue is how to select the best scan positions for the PDM^2 sensor, which can further speed up the perception process while satisfying the uncertainty requirement through active perception. We develop an optimization-based approach by increasing the information gain (IG) and prioritizing the boundary points under a preset travel distance constraint that is designed to ensure trajectory smoothness.

(a) (b)

Fig. 1. Shape reconstruction results overlaid as red projected vertices on the image: (a) using a camera-only method and (b) our sensor fusion method.

Fig. 1 (b) shows the successful shape reconstruction as a result of our algorithms. Further experimental results show that our algorithms increase the intersection over union (IoU) metric from 0.73 to 0.96, a significant increase in reconstruction quality over camera-only methods.

II. Related Works

Our approach builds on the following related topics including 1) the development of the PDM^2 sensor, 2) challenges in transparent object reconstruction, 3) using GP to estimate unknown shapes, 4) using IG to assess uncertainty, and 5) active perception.

We have developed a PDM^2 sensor, previously named a dual-mode and dual-sensing mechanism (DMDSM) sensor, to sense object material, interior structure, and distance right before a robot touches an object of interest. It is a versatile sensor mounted on the robot fingertip to provide real-time pre-touch information for optically and acoustically challenged objects in environments where little prior information is known. The sensor combines pulse echo ultrasound (US) and optoacoustic (OA) modalities and has evolved through four generations of iterative designs [1], [2], [4]–[7]. In fact, the latest generation design that employs a custom acoustic-to-optical receptor to significantly improve signal-to-noise ratio is also accepted by this conference [8]. Using the sensor, we have also developed the material mapping algorithm and

F. Guo, S. Xie, D. Wang, and D. Song are with CSE Department, Texas A&M University, College Station, TX 77843, USA. D. Song are also with Department of Robotics, Mohamend Bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE. Email: dezhen.song@mbzuai.ac.ae.

C. Fang and J. Zou are with ECE Department, Texas A&M University, College Station, TX 77843, USA, Email: junzou@tamu.edu.

This work was supported in part by National Science Foundation under IIS-2119549 and NRI-1925037, by Amazon Research Award, by GM/SAE AutoDrive Challenge, and by MBZUAI.

tested it in a compact scanning system [3], [7]. This paper explores how to fuse this new sensor with camera images to improve system accuracy and speed in object reconstruction, which is probably the most likely use case scenario in future applications.

Realizing the importance of handling transparent objects, many recent works focus on improving camera-based object reconstruction method. These methods can be classified as either model-based optimization methods [9]–[12] or learning-based methods [13]–[19]. However, because light paths are distorted by reflection and refraction, their reconstruction quality suffers from point-cloud imperfections, which include point-wise noise, uneven distribution, missing points, misalignment, and a significant number of outliers [20]. That explains why the silhouette-based reconstruction method utilizing the Segment Anything Model [21] suffers from imprecise segmentation when applied to transparent objects. With the advent of the radiance field, NeRF and Gaussian Splatting provide a robust object or scene representation [22]–[29]. Evo-NeRF [23] provides an incremental NeRF optimization with active perception, but focuses on grasping instead of a high-fidelity reconstruction. While these new methods aim at handling reflectivity or transparency shows advancements, they are clearly limited by the sensing modality, as they often require extra background configurations under particular lighting and transparency setups, and their training process is computation-intensive, which limits their field applicability. In this work, with the new PDM^2 sensor, it is a natural solution to employ a sensor fusion approach.

GP [30], as a nonparametric model, is a great choice to model objects of unknown shape that can quantify uncertainty and handle noise in the estimation process. Many existing works exploit the advantage. Gandler *et al.* [31] apply GP as an implicit surface representation to facilitate a sensor fusion approach using a camera and a tactile sensor. Another closely related work is ShapeMap-3D [32], which combines a GelSight™ tactile sensor and a depth camera using an incremental shape mapping approach to reconstruct the shape of the object. Both works inspire our approach. Due to the characteristics of the PDM^2 sensor, our algorithm has to be an incremental and iterative update instead of the full computation at once like [31] for better efficiency. Also, unlike the 2D Gelsight sensor, the PDM^2 sensor is a point sensor, planning its scanning positions/trajectory to balance speed versus accuracy is a unique problem.

When selecting the best PDM^2 sensor scan positions, we employ IG to measure the reduction in the uncertainty in the reconstruction process. This is inspired by the information-theoretic exploration with Bayesian optimization (BO) [33]. IG computation is often performed in combination with an occupancy grid map with independent cells. In our case, because the continuous object contour leads to highly correlated nearby points, our algorithm selects candidate positions over an irregular lattice instead of a fixed 2D grid as in Yang *et al.* [34]'s work where they explicitly calculate IG of the sampled actions in the 2D grid map and utilizes Bayesian Kernel Inference (BKI) method to estimate the IG and its corresponding uncertainty. IG-based optimization is also developed to facilitate sensor placement on a 2D grid [35]. In addition, we employ distributed GP [36] to efficiently update the fused IG, which paves the way for an iterative approach to planning the position of the scan.

The scanning position planning for the PDM^2 sensor is an active perception problem. When the planned sensor is a camera, the active perception problem is also called an active view planning problem [37]. Compared to passive view planning, which estimates the view sequence from the prior and then fixes the sequence for the overall perception process, recent active view planning research focuses on active / interactive perception [38]–[40] and informative path planning [34], [41], [42]. Active view planning methods progressively update view sequence after gaining more information. For view selection, Mao and Xiao [43] average the uncertainty over the points at the intersection of the target plane and the isosurface. Since our PDM^2 sensor is a point sensor, it determines that active scan planning has to take into account IG, travel distance, and scanning coverage, which is different from view planning for a camera.

III. Problem Formulation

A. Nomenclature

Before we define our problem, common variables are defined as follows.

- $\{\mathbf{0}\}$ denotes the world frame which is a right-handed 3D Euclidean coordinate system.
- $\mathbf{x}$ is a point position in $\{\mathbf{0}\}$, $\mathbf{x} \in \mathbb{R}^3$ with the corresponding covariance matrix Σ.
- $\mathcal{P}_c$, $\mathcal{P}_d$ represent the 3D point clouds in $\{\mathbf{0}\}$ reconstructed from the camera and the PDM^2 sensor, respectively. $\mathcal{P}_c := \{\mathbf{x}_i\}_{i=1:n_c}$, where n_c is the number of points in the point cloud. The error of $\mathbf{x} \in \mathcal{P}_c$ is often more than a millimeter due to transparency issues, which is reflected by the covariance matrix set $\mathbf{\Sigma}_c = \{\Sigma_i\}_{i=1:n_c}$ where $\Sigma_i \in \mathbb{R}^{3\times3}$ is the covariance matrix for $\mathbf{x}_i$. Similarly, we have $\mathcal{P}_d := \{\mathbf{x}_i\}_{i=1:n_d}$ and $\mathbf{\Sigma}_d = \{\Sigma_i\}_{i=1:n_d}$, where the difference is that the accuracy of $\mathbf{x} \in \mathcal{P}_d$ reaches sub-millimeter level.
- $F(\mathbf{x})$ is the signed distance from $\mathbf{x}$ to the object surface, which is positive for inner points and negative for outer points. Since $\{\mathbf{x} : F(\mathbf{x}) = 0\} \subset \mathbb{R}^3$ defines the surface of the scanned object, $F(\mathbf{x})$ is also called the implicit function of the surface.
- $\sigma_{\text{F}}^2(\mathbf{x})$ is the variance of the signed distance $F(\mathbf{x})$ at $\mathbf{x}$.

B. Software Diagram

Fig. 2 shows the overall software diagram. We first employ the Gaussian process (GP) method to reconstruct the implicit function of the object surface $F_0(\mathbf{x})$ from the point cloud constructed from the camera image $\mathcal{P}_c$. $\mathbf{x}$'s covariance set $\mathbf{\Sigma}_c$ characterizes its uncertainty. Due to the high uncertainty in $\mathcal{P}_c$, we cannot trust $\{\mathbf{x} : F_0(\mathbf{x}) = 0\}$ as a reliable representation of the surface of the object. To address this problem, we incrementally and iteratively scan more surface points using the PDM^2 sensor in batches. We know that

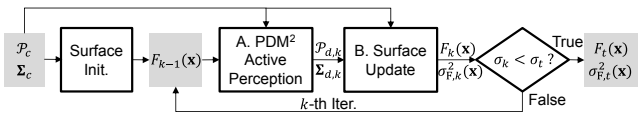


Fig. 2. Software diagram. "Init." means initialization and "Iter." means iteration. The subscript k is the iteration index and the subscript t means termination.

combining newly scanned points with the existing point cloud increases the accuracy of object reconstruction because the points scanned by the PDM^2 sensor are more accurate. As a result, at the k-th iteration, it is necessary to select a batch of new points from the current surface to form the next trajectory to guide the scanning of the new batch of points, which is an active perception problem in nature (Box A in Fig. 2). After scanning, we obtain the newly collected point cloud $\tilde{\mathcal{P}}_{d,k}$ and the updated overall PDM^2-based point cloud $\mathcal{P}_{d,k} := \mathcal{P}_{d,k-1} \cup \tilde{\mathcal{P}}_{d,k}$ which are used to estimate the implicit surface $F_k(\mathbf{x})$ and its variance (Box B in Fig. 2). This GP-based heterogeneous sensor fusion reduces the variance of all surface points. If the maximum variance σ_k is below a preset threshold σ_t, then the pipeline is successfully terminated. Otherwise, we repeat the algorithm by planning more scans for the PDM^2 sensor.

C. Problem Definition

The above pipeline contains two problems which are the surface reconstruction problem and the PDM^2 sensor active perception problem corresponding to Boxes A and B in Fig. 2, respectively. We define them as follows.

Definition 1 (Surface Reconstruction): At the k-th iteration, given point clouds $\mathcal{P}_c$ and $\mathcal{P}_{d,k}$ with their corresponding covariance sets $\mathbf{\Sigma}_c$ and $\mathbf{\Sigma}_{d,k}$, construct the implicit surface function $F_k(\mathbf{x})$ and obtain its variance $\sigma^2_{F,k}(\mathbf{x})$.

Definition 2 (Active Perception for the PDM^2 Sensor): At the k-th iteration, given the current point clouds $\mathcal{P}_c$ and $\mathcal{P}_{d,k-1}$ with covariance matrix sets $\mathbf{\Sigma}_c$ and $\mathbf{\Sigma}_{d,k-1}$, planning for scanning positions for the PDM^2 sensor to obtain new points $\tilde{\mathcal{P}}_{d,k}$ to update $\mathcal{P}_{d,k}$.

IV. Algorithms

Now we present two algorithms to solve the two problems in Secs. IV-A and IV-B, respectively.

A. Surface Reconstruction

Fig. 2 shows that the surface reconstruction appears in two places in the pipeline. The first place is initialization, where we estimate an initial noisy implicit surface function directly from the image point cloud. The second place is after new PDM^2 points are obtained, where we update the surface implicit function based on both the image point cloud and the newly obtained PDM^2 points. Both places use the same method with different corresponding covariance matrices, which we will explain as follows.

1) Implicit Surface Function Modeling: We employ GP as an implicit function to represent the 3D surface and its uncertainty. For all points in the input point cloud, their corresponding signed distances can be modeled using a GP, which is a finite collection of random variables that follow a joint Gaussian distribution [44]. The GP can be written as $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$, where $m(x) = \mathbb{E}[f(\mathbf{x})]$ and $k(\mathbf{x}, \mathbf{x}') = \mathbb{E}\left[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))\right]$ for any point $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^3$. The point $\mathbf{x}$ used to reconstruct the surface comes from two sources. The image points in $\mathcal{D}_c = \{\mathbf{x}_i, v_i\}_{i=1}^{n_c}$ provide the initial observations needed for surface reconstruction, where v_i is the observed value of the signed distance value for point $\mathbf{x}_i$ with a positive value indicating that the point is inside the surface and zero means on the surface. More exactly, v_i is modeled as follows,

$$v_i = f(\mathbf{x}_i) + e(\mathbf{x}_i), \tag{1}$$

where random error $e(\mathbf{x}_i) \sim \mathcal{N}(0, a_i^2)$ follows a zero-mean Gaussian distribution with a_i^2 being its variance [45]. For the i-th point in $\mathcal{P}_c \cup \mathcal{P}_{d,k}$, we approximate the variance $a_i^2 = \frac{1}{3}\mathrm{tr}\,(\Sigma_i)$, where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix and $\Sigma_i \in \mathbf{\Sigma}_c \cup \mathbf{\Sigma}_{d,k}$.

The image point cloud $\mathcal{D}_c$ is used throughout the surface reconstruction. In the update stage in k-iteration, new data from the PDM^2 sensor arrive, which is $\mathcal{D}_d = \{\mathbf{x}_i, v_i\}_{i=1}^{n_{d,k}}$. We define the input point cloud as $\mathcal{X} := \mathcal{P}_c \cup \mathcal{P}_{d,k}$ with $n := |\mathcal{X}|$, where $|\mathcal{X}|$ is the cardinality of $\mathcal{X}$. $\mathcal{P}_{d,0} = \emptyset$ at the initialization stage. $\mathcal{X}$ is the main input to the surface construction problem.

To simplify notation, let us define the row vector and matrix representations as $[w_i]_{i=1}^n := [w_1, ..., w_n]$ and $[\mathbf{w}]_{i=1}^n := [\mathbf{w}_1, ..., \mathbf{w}_n]$, respectively, where w is a scalar template and $\mathbf{w}$ is a column vector template, and w and $\mathbf{w}$ will be replaced by respective notation later. Then, we write the joint distribution of the observations $\mathbf{v}^\mathsf{T} = [v_i]_{i=1}^n$ at $\mathcal{X}$ and the signed distance values $\mathbf{v}^*$ at the target test point set $\mathcal{X}^*$ under the prior as

$$\begin{bmatrix} \mathbf{v} \\ \mathbf{v}^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \Sigma_{\mathbf{XX}} & K(\mathbf{X}, \mathbf{X}^*) \\ K(\mathbf{X}^*, \mathbf{X}) & K(\mathbf{X}^*, \mathbf{X}^*) \end{bmatrix} \right), \tag{2}$$

where the mean vector $\mathbf{b}^\mathsf{T} = [b(\mathbf{x}_i)]_{i=1}^n$, the zero vector $\mathbf{0}^\mathsf{T} = [0]_{i=1}^{n^*}$, the observation input matrix $\mathbf{X} = [\mathbf{x}_i]_{i=1}^n$, $\mathbf{x}_i \in \mathcal{X}$, the predictive input matrix $\mathbf{X}^* = [\mathbf{x}_i^*]_{i=1}^{n^*}$, $\mathbf{x}_i^* \in \mathcal{X}^*$, $\Sigma_{\mathbf{XX}} = K(\mathbf{X}, \mathbf{X}) + \mathbf{A}$, the Gram matrix $K(\mathbf{X}_1, \mathbf{X}_2)$ for $\mathbf{X}_1, \mathbf{X}_2 \in \{\mathbf{X}^*, \mathbf{X}\}$ are evaluated at all pairs of entries of $\mathbf{X}_1$ and $\mathbf{X}_2$, $K(\mathbf{X}_1, \mathbf{X}_2) = [k(\mathbf{x}_{1,i}, \mathbf{x}_{2,j})]_{i \in [1..|\mathbf{X}_1|], j \in [1..|\mathbf{X}_2|]}$, the random noise covariance matrix $\mathbf{A} = \mathrm{diag}(a_1^2, a_2^2, ..., a_n^2)$, and $n^* := |\mathcal{X}^*|$.

Given an input point cloud $\mathcal{X}$, its observation vector $\mathbf{v}$, the mean vector $\mathbf{b}$, the conditional distribution of the signed distance values $\mathbf{v}^*$ at the unobserved point set $\mathcal{X}^*$ is a Gaussian as below

$$\mathbf{v}^* | \mathbf{X}^*, \mathbf{X}, \mathbf{v}, \mathbf{b} \sim \mathcal{N}\left(\mu_{\mathcal{X}^*|\mathcal{X}}, \Sigma_{\mathcal{X}^*|\mathcal{X}}\right), \tag{3}$$

with conditional mean $\mu_{\mathcal{X}^*|\mathcal{X}}$ and covariance $\Sigma_{\mathcal{X}^*|\mathcal{X}}$:

$$\mu_{\mathcal{X}^*|\mathcal{X}} = K(\mathbf{X}^*, \mathbf{X})\Sigma_{\mathbf{XX}}^{-1}(\mathbf{v} - \mathbf{b}), \tag{4}$$

$$\Sigma_{\mathcal{X}^*|\mathcal{X}} = K(\mathbf{X}^*, \mathbf{X}^*) - K(\mathbf{X}^*, \mathbf{X})\Sigma_{\mathbf{XX}}^{-1}K(\mathbf{X}^*, \mathbf{X})^{\mathsf{T}}. \tag{5}$$

Note that when $n^* = 1$, $\Sigma_{\mathcal{X}^*|\mathcal{X}}$ is a scalar, so we define $\sigma_{\mathcal{X}^*|\mathcal{X}} := \Sigma_{\mathcal{X}^*|\mathcal{X}}$. However, the basic GP model of (4) and (5) has a time complexity of $O(n^3)$ for n points for surface reconstruction in each iteration. When new PDM^2 sensor scan points are obtained, recomputing the entire GP model is too slow. We need a more efficient GP update process.

2) Surface Implicit Function Update: We employ the distributed GP [36] to address the problem, which partitions $\mathcal{X}$ into M independent GP experts. Therefore, we only need to update the GP expert that is affected by the new points. The method is called mGP as opposed to the original GP. The posterior mean and variance of $\mathbf{x}^*$ by the product of the GP experts are given by

$$\mu^o_{\{\mathbf{x}^*\}|\mathcal{X}} = \left(\sigma^o_{\{\mathbf{x}^*\}|\mathcal{X}}\right)^2 \sum_{m=1}^{M} \sigma^{-2}_{\{\mathbf{x}^*\}|\mathcal{X}_m}\mu_{\{\mathbf{x}^*\}|\mathcal{X}_m}, \tag{6}$$

$$\left(\sigma^o_{\{\mathbf{x}^*\}|\mathcal{X}}\right)^{-2} = \sum_{m=1}^{M} \sigma^{-2}_{\{\mathbf{x}^*\}|\mathcal{X}_m}, \tag{7}$$

where m is the group or Gaussian expert index. With the distributed GP, the time complexity to update the posterior distribution in a new iteration is dropped from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^3/M^2)$ in the worst-case scenario, which happens when all GP experts need to be changed. The complexity can be reduced to $\mathcal{O}(n^3/M^3)$ if all newly scanned points are assigned to the same GP expert. This is possible because the newly scanned points in each iteration are usually close to each other, which will be discussed further in Sec. IV-B.

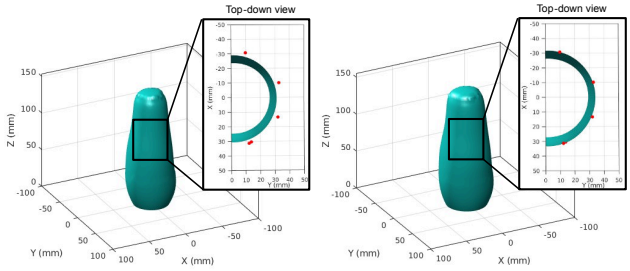
In addition, the reconstruction results in (6) and (7) also depend on the partition of points between different GP experts. An intuitive partition method should attempt to assign adjacent points to the same GP expert to ensure surface consistency. Therefore, to partition $\mathcal{X}$ into $\{\mathcal{X}_m\}_{m=1..M}$, we build an undirected weighted graph $\mathcal{G}$ with node set $\mathcal{X}$ and adjacency matrix $\mathbf{C}$ for all elements in $\mathcal{X}$ and use the METIS method [46] to partition the graph, which minimizes edge cuts and leads to a balanced and efficient graph partition.

For the entry at p-th row q-th column, $c_{p,q} \in \mathbf{C}$ is denoted as

$$c_{p,q} = \begin{cases} \frac{\sigma_{p,q}}{\sigma_p\sigma_q}, & \text{if } (p \neq q) \wedge (\frac{\sigma_{p,q}}{\sigma_p\sigma_q} \geq c_t), \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

where $\sigma_{p,q}$ is the entry of $\mathbf{\Sigma_{XX}}$ at p-th row and q-th column, $\sigma_p = \sqrt{\sigma_{p,p}}$, $\sigma_q = \sqrt{\sigma_{q,q}}$, and c_t is a threshold to remove the edges with low correlation. Note that the partitioning process only needs to be computed at the initialization stage. Any newly scanned point $\mathbf{x}_d \in \mathcal{P}_d$ is assigned to the same GP expert as that of the closest point.

3) Bias Removal: Eq. (1) assumes zero mean for the noise distribution. This zero-bias assumption may not hold for points in $\mathcal{P}_c$ due to shape ambiguity due to light reflection and refraction introduced by transparency objects.



(a) (b)

Fig. 3. Surface reconstruction from $\mathcal{P}_c$ without and with bias removal for (a) and (b), respectively. The red points come from $\mathcal{P}_d$ as the reference points. Without considering the bias, the reconstructed surface shrinks inward in this case. After removing the bias, the surface tends to be expand to approach the reference points.

Fig. 3 illustrates this phenomenon. On the other hand, points scanned by the PDM^2 sensor are unbiased due to its sensing mechanism. For $\mathbf{x}_i$, let $b(\mathbf{x}_i)$ be its bias, we have

$$b(\mathbf{x}_i) = \begin{cases} \frac{1}{n_d}\sum_{d=1}^{n_d} -\mu^o_{\{\mathbf{x}_d\}|\mathcal{P}_c} & \text{if } (\mathbf{x}_i \notin \mathcal{P}_{d,k}), \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

where $\mathbf{x}_d \in \mathcal{P}_{d,k}$, $n_d = |\mathcal{P}_{d,k}|$, and $\mu^o_{\{\mathbf{x}_d\}|\mathcal{P}_c}$ is obtained from (6).

To remove bias, we have $F(\mathbf{x}) = \mu^o_{\{\mathbf{x}\}|\mathcal{X}}$ and $\sigma^2_{\mathrm{F}}(\mathbf{x}) = \left(\sigma^o_{\{\mathbf{x}\}|\mathcal{X}}\right)^2$. After the update, we check if the maximum variance of all points in the k iteration σ_k falls below the threshold σ_t. If so, we terminate the algorithm. If not, we need to plan for more scanning for the PDM^2 sensor, which leads to the active perception problem.

B. Active Perception for the PDM^2 Sensor

The PDM^2 sensor complements the camera in perceiving the transparent object shape. Unlike a camera, the PDM^2 sensor scanning is relatively slow because it is a point sensor. The readings from the PDM^2 sensor are more accurate. Therefore, strategic planning for PDM^2 sensor scanning is important, which leads to the unique problem of active perception. To address the problem and note that the overall goal is to reduce reconstruction uncertainty, we devise an optimization over a sampling-based planning approach that 1) maximizes information gain (IG) and 2) rewards more on boundary candidates between low- and high- variance regions to reduce the ineffective back-and-forth movements over scanning point choices and trajectory generation. The optimal choice is bounded below a preset trajectory length to accommodate the batch scanning requirement.

1) Sampling-based Candidate Solutions: Since the resulting trajectory of our problem must be located on the surface described by the implicit shape function $F_{k-1}(\mathbf{x})$ at the k-th iteration, we sample $\{\mathbf{x} : F_{k-1}(\mathbf{x}) = 0\}$ to generate a set of candidate solutions $\mathcal{U}_k$ to reduce planning time. The sampling is done by applying the marching cube algorithm [47]. $300 \leq |\mathcal{U}_k| \leq 400$ is the sampling setup because that provides sufficient candidate resolution for common household items. The choice of scan position is to be obtained by

solving an iterative optimization problem over $\mathcal{U}_k$. Therefore, some points in $\mathcal{U}_k$ may have been selected in the previous iteration. We abuse the notation of $\mathcal{U}_k$ by assuming that it is the remaining set of candidate solutions at the current iteration. $\mathcal{U}_k$ is updated after each iteration.

2) Using Information Gain to Measure Uncertainty Reduction: The first component of the objective function is IG because IG considers the entropy prediction quality over the space of interest instead of just the selected points [48]. We define the overall point set $\mathcal{W}_k := \mathcal{U}_k \cup \mathcal{P}_c \cup \mathcal{P}_{d,k-1}$ in the k-th iteration. For simplicity, we omit the subscript of the iteration index k and have $\mathcal{W} := \mathcal{W}_k$ and $\mathcal{U} := \mathcal{U}_k$. A simple approach is to select a subset $\mathcal{V} \subseteq \mathcal{U}$ to maximize IG

$$\mathcal{V}^* = \underset{\mathcal{V}\subseteq\mathcal{U}:|\mathcal{V}|=n_b}{\operatorname{argmax}} H(\mathbf{V}_{\mathcal{W}\setminus\mathcal{V}}) - H(\mathbf{V}_{\mathcal{W}\setminus\mathcal{V}}|\mathbf{V}_{\mathcal{V}}) \tag{10}$$

where n_b is batch size, $H(\mathbf{V}_{\mathcal{W}\setminus\mathcal{V}})$ is the differential entropy of the unobserved points $\mathcal{W} \setminus \mathcal{V}$, $H(\mathbf{V}_{\mathcal{W}\setminus\mathcal{V}}|\mathbf{V}_{\mathcal{V}})$ is the conditional differential entropy of the unobserved points $\mathcal{W} \setminus \mathcal{V}$ after observing points $\mathcal{V}$, and $\mathbf{V}_{\mathcal{V}}$ and $\mathbf{V}_{\mathcal{W}\setminus\mathcal{V}}$ refer to the vector of GP random variables corresponding to $\mathcal{V}$ and $\mathcal{W} \setminus \mathcal{V}$, respectively. Since directly solving this optimization problem is NP-complete, we use the approximation algorithm in [35] to greedily select the j-th point of the batch $\mathbf{x}_j$ from the current candidate set $\mathcal{U} \setminus \tilde{\mathcal{V}}_j$, where the current solution set $\tilde{\mathcal{V}}_j := \{\mathbf{x}_i\}_{i=1..j-1}$ is obtained by solving

$$\mathbf{x}_j = \underset{\mathbf{x}_g\in\mathcal{U}\setminus\tilde{\mathcal{V}}_j}{\operatorname{argmax}} \frac{\sigma_{\{\mathbf{x}_g\}|\mathcal{W}\setminus\mathcal{R}_{j,g}}}{\sigma_{\{\mathbf{x}_g\}|\mathcal{R}_{j,g}}} \tag{11}$$

where a scalar $\sigma_{\mathcal{X}^*|\mathcal{X}} := \Sigma_{\mathcal{X}^*|\mathcal{X}}$ if $|\mathcal{X}^*| = 1$, and $\mathcal{R}_{j,g} := \mathcal{U} \setminus \{\tilde{\mathcal{V}}_j \cup \{\mathbf{x}_g\}\}$. It is noted that the numerator and denominator in (11) are scalars and the low-cost surrogate can be calculated from (7).

It is not difficult to see that maximizing IG alone cannot guarantee a proper trajectory because the trajectory inevitably travels back and forth to find points with high IG values. It leads to an inefficient solution in application. We should evaluate a candidate solution from a motion efficiency perspective in addition to IG.

3) Improve Motion Efficiency by Prioritizing Boundary Points: During the PDM^2 sensor scanning process, the newly scanned points has low variance and they help its immediate neighboring points' variance to be reduced. If a candidate scanning point $\mathbf{x}_h \in \mathcal{U}$'s variance is below a given uncertainty threshold σ_t, then it is a low-variance point described by indicator function $\mathbb{1}_{\text{LV}}$,

$$\mathbb{1}_{\text{LV}}(\mathbf{x}_h) = \begin{cases} 1, & \text{if } \sigma_{\{\mathbf{x}_h\}|\mathcal{W}\setminus\mathcal{V}} < \sigma_t \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Therefore, all points on the surface can be classified into two categories according to the values of $\mathbb{1}_{\text{LV}}$, leading to a division between regions of low and high variances. The idea is to prioritize the points on the boundary so that the resulting scanning movements do not jump back and forth inefficiently. To identify existing boundary points, we introduce the following spherical in-annulus condition for $\mathbf{x}_g \in \mathcal{U} \setminus \tilde{\mathcal{V}}_j$ as follows,

$$\mathbb{1}_{\text{in}}(\mathbf{x}_g, \mathbf{x}_h) = \begin{cases} 1, & \text{if } \tau_1 < \|\mathbf{x}_g - \mathbf{x}_h\|_2^2 < \tau_2 \\ 0, & \text{otherwise,} \end{cases} \tag{13}$$

where τ_1 and τ_2 define the inner and outer bounds of the spherical annulus, respectively. τ_2 determines the neighboring range, while τ_1 avoids the influence of points that are too close. Too many high-variance points close together can unduly reduce boundary priority as we want the neighbors to spread out. For the given candidate point $\mathbf{x}_g$, we are interested in the ratio η_g of the low variance points among all in-annulus neighbors,

$$\eta_g = \begin{cases} \frac{\sum_{\mathbf{x}_h\in\mathcal{U}} \mathbb{1}_{\text{in}}(\mathbf{x}_g,\mathbf{x}_h)\cdot\mathbb{1}_{\text{LV}}(\mathbf{x}_h)}{\sum_{\mathbf{x}_h\in\mathcal{U}} \mathbb{1}_{\text{in}}(\mathbf{x}_g,\mathbf{x}_h)}, & \text{if } \sum\limits_{\mathbf{x}_h\in\mathcal{U}} \mathbb{1}_{\text{in}(\mathbf{x}_g,\mathbf{x}_h)} > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

Since we want to reward those candidates with η_g greater than the given threshold η_t, we define the following indicator function,

$$\mathbb{1}_{\text{RLV}}(\mathbf{x}_g) = \begin{cases} 1, & \text{if } \eta_g > \eta_t \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Now we can modify the optimization formulation in (11) by incorporating boundary prioritization as follows.

$$\mathbf{x}_j = \underset{\mathbf{x}_g\in\mathcal{U}\setminus\tilde{\mathcal{V}}_j}{\operatorname{argmax}} \frac{\sigma_{\{\mathbf{x}_g\}|\mathcal{W}\setminus\mathcal{R}_{j,g}}}{\sigma_{\{\mathbf{x}_g\}|\mathcal{R}_{j,g}}} + \lambda\mathbb{1}_{\text{RLV}}(\mathbf{x}_g), \tag{16}$$

$$\text{such that } ||\mathbf{x}_j - \mathbf{x}_{j-1}||_2 \leq \gamma, \tag{17}$$

where hyper-parameter λ determines how much we want to emphasize boundary priority. Eq. (16) still has a problem because it does not prevent the trajectory from jumping back and forth between boundary points. To deal with this, we need to regulate the trajectory length in (17) where γ is a hyperparameter to set the upper limit of the travel length between two neighboring scan points.

Eqs. (16) and (17) provide an approximate solution to our active perception problem. Each time we solve the optimization problem, we can obtain a scan point $\mathbf{x}_j$. We remove $\mathbf{x}_j$ from $\mathcal{U}$. Since the PDM^2 sensor scans object surface in batches, we repeatedly solve the optimization problem n_b times to obtain the planned trajectory. By executing this trajectory, we can obtain a new point cloud $\tilde{\mathcal{P}}_{d,k}$. The total PDM^2 points $\mathcal{P}_{d,k} = \mathcal{P}_{d,k-1} \cup \tilde{\mathcal{P}}_{d,k}$ is used for surface reconstruction in the next iteration.

V. Experiments

A. System Configuration and Data Collection

We have used the scanning platform in [3] to collect test data from an Intel™ RealSense depth camera D435 and our PDM^2 sensor. The only modification we have made is to mount the two sensors together. On the software side, the algorithm is implemented in Matlab and executed on an i7-13700K CPU running Ubuntu 20.04 system. The squared exponential kernel function is used for the GP. Limited by SNR, this sensor takes 20 s to scan a point.

(a) (b) (c) (d) (e) (f)

Fig. 4. Transparent objects used in the experiment. (a) Juice cup. (b) Water bottle. (c) Coffee bottle. (d) Red wine cup. (e) Cola bottle. (f) Soup bottle. (a-d) are made of glass while (e,f) are made of plastics.

Fig. 4 shows the six transparent objects used in our experiments. For each object, we place it inside the scanning system. The camera is placed in three different locations to ensure complete coverage of the object. At each location, the camera captures the object by rotating it in 5-degree increments until it returns to its original position. Next, we reconstruct the initial reconstruction results from the images using the structure-from-motion method [49]. We then apply our algorithms to iteratively generate a batch of scanning points. Fig. 5 shows the typical scan and reconstruction results for a glass bottle. It is clear that the initial image-based object reconstruction has a very large error (leftmost shape in the middle row). Our heterogeneous sensor fusion algorithms iteratively improve object reconstruction quality.

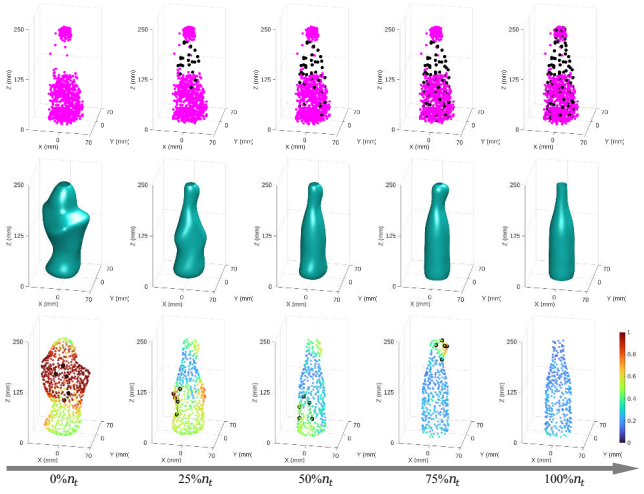


Fig. 5. Shape reconstruction and active perception planning for a glass bottle. Top: pink points are the point cloud from image points $\mathcal{P}_c$ while black points are scanned by the PDM^2 sensor. Middle: bottle reconstruction result. Bottom: Black dots indicate scanning trajectories, and the standard deviation of the reconstructed point are colored using the right side spectrum which has a normalized value ranging from 0 to 1 for illustration purpose.

B. Metrics and Results

Since the exact shape of the test objects may not be available, we employ two metrics to measure the quality of the reconstruction. The first metric is the intersection-over-union [50] between the projected shape from the reconstruction result $\mathcal{W}_{\text{proj}}$ and the manually-labored ground-truth 2D silhouette $\mathcal{W}_{\text{gt}}$ in the images collected,

$$\text{IoU} = \frac{|\mathcal{W}_{\text{proj}} \cap \mathcal{W}_{\text{gt}}|}{|\mathcal{W}_{\text{proj}} \cup \mathcal{W}_{\text{gt}}|}. \tag{18}$$

The region covered by the red lattice in Fig. 1 are examples of $\mathcal{W}_{\text{proj}}$. Then, we evaluate the reconstruction results from the overall $\overline{\text{IoU}}$, calculated as the average of IoUs from 4 orthogonal perspectives.

The second metric is the ratio κ_k of low-variance points on the most recently sampled candidate points $\mathcal{U}_k$, which is a random lattice that covers the entire object. We have

$$\kappa_k = \frac{\sum_i^{|\mathcal{U}_k|} \mathbb{1}_{\text{RA}}(\mathbf{x}_i)}{|\mathcal{U}_k|}, \text{and } \mathbb{1}_{\text{RA}}(\mathbf{x}_i) = \begin{cases} 1, & \text{if } \sigma_{\text{F}}(\mathbf{x}_i) < \sigma_t, \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

For simplification, we refer to κ_k as κ if k is not the focus. The experimental results are shown in Table I. Note that the overall number of points scanned by our PDM^2 sensor is $n_t = |\mathcal{P}_{d,k}|$ after the variances of all points are below σ_t. In the object shape reconstruction, we gradually increase the number of the PDM^2 points from 0, which means camera only, to 100%n_t points to observe how $\overline{\text{IoU}}$ and κ change. Tab. I shows that our heterogeneous sensor fusion approach significantly increases reconstruction quality. On average, $\overline{\text{IoU}}$ increases from 0.73 to 0.96, indicating a substantial improvement. Note that the $\overline{\text{IoU}}$ cannot reach the maximum value of 1 because there is surface smoothness and manual labeling error.

TABLE I

SHAPE RECONSTRUCTION RESULTS FOR OBJECTS IN FIG. 4 WITH DIFFERENT PARAMETER CONFIGURATIONS

Obj.	n_t	$\overline{\text{IoU}}$ and (κ)				
		Cam. Only	25%n_t	50%n_t	75%n_t	100%n_t
(a)	80	.67 (0%)	.88 (37%)	.97 (84%)	.97 (94%)	**.97 (100%)**
(b)	90	.50 (0%)	.87 (28%)	.90 (71%)	.91 (92%)	**.95 (100%)**
(c)	70	.80 (0%)	.93 (44%)	.96 (95%)	.96 (96%)	**.97 (100%)**
(d)	65	.86 (0%)	.91 (22%)	.94 (54%)	.97 (85%)	**.97 (100%)**
(e)	85	.74 (2%)	.83 (25%)	.93 (75%)	.94 (99%)	**.94 (100%)**
(f)	50	.79 (0%)	.84 (34%)	.89 (77%)	.91 (98%)	**.93 (100%)**
Avg.	73	.73 (0%)	.88 (32%)	.93 (76%)	.94 (94%)	**.96 (100%)**

VI. CONCLUSION AND FUTURE WORK

We reported a sensor fusion algorithm to tackle the challenging task of reconstructing the shape of transparent household items such as glass bottles or cups, because traditional camera-based reconstruction is often not reliable due to distorted light paths. We combined the image input with the point-wise scan from our PDM^2 sensor by developing a distributed GP-based shape reconstruction method and an active perception method based on maximizing IG and prioritizing boundary points while considering the travel distance constraint. The overall algorithm was tested under a custom scanning platform with six transparent objects. The results of the experiment are satisfactory.

In the future, we will further develop the grasping algorithm based on the reconstruction result. We will combine the shape and material types of the PDM^2 sensor for better planning for the grasping of delicate and transparent objects.

ACKNOWLEDGMENT

We are grateful to K. Goldbeg, A. Kingery, C. Hu, and Z. Sun for their inputs and feedback.

REFERENCES

[1] C. Fang, D. Wang, D. Song, and J. Zou, "The second generation (g2) fingertip sensor for near-distance ranging and material sensing in robotic grasping," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 1506–1512.

[2] C. Fang, S. Li, D. Wang, F. Guo, D. Song, and J. Zou, "The third generation (g3) dual-modal and dual sensing mechanisms (dmdsm) pretouch sensor for robotic grasping," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 1731–1736.

[3] F. Guo, S. Xie, D. Wang, C. Fang, J. Zou, and D. Song, "A pretouch perception algorithm for object material and structure mapping to assist grasp and manipulation using a dmdsm sensor," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 6831–6838.

[4] C. Fang, D. Wang, D. Song, and J. Zou, "Toward fingertip non-contact material recognition and near-distance ranging for robotic grasping," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4967–4974.

[5] ——, "Fingertip non-contact optoacoustic sensor for near-distance ranging and thickness differentiation for robotic grasping," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 894–10 899.

[6] ——, "Fingertip pulse-echo ultrasound and optoacoustic dual-modal and dual sensing mechanisms near-distance sensor for ranging and material sensing in robotic grasping," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 14 105–14 111.

[7] D. Wang, F. Guo, C. Fang, J. Zou, and D. Song, "Design of an object scanning system and a calibration method for a fingertip-mounted dual-modal and dual sensing mechanisms (dmdsm)-based pretouch sensor for grasping," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 341–347.

[8] C. Fang, Z. Yan, F. Guo, S. Li, D. Song, and J. Zou, "A full-optical pre-touch dual-modal and dual-mechanism (pdm2) sensor for robotic grasping," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025.

[9] A. C. Öztireli, G. Guennebaud, and M. Gross, "Feature preserving point set surfaces based on non-linear kernel regression," in *Computer graphics forum*, vol. 28, no. 2. Wiley Online Library, 2009, pp. 493–501.

[10] M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 32, no. 3, pp. 1–13, 2013.

[11] M. Atzmon and Y. Lipman, "Sal: Sign agnostic learning of shapes from raw data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2565–2574.

[12] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, "Implicit geometric regularization for learning shapes," *arXiv preprint arXiv:2002.10099*, 2020.

[13] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.

[14] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3d reconstruction in function space," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4460–4470.

[15] C. Jiang, A. Sud, A. Makadia, J. Huang, M. Nießner, T. Funkhouser *et al.*, "Local implicit grid representations for 3d scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6001–6010.

[16] G. Sharma, D. Liu, S. Maji, E. Kalogerakis, S. Chaudhuri, and R. Měch, "Parsenet: A parametric surface fitting network for 3d point clouds," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*. Springer, 2020, pp. 261–276.

[17] P. Erler, P. Guerrero, S. Ohrhallinger, N. J. Mitra, and M. Wimmer, "Points2surf learning implicit surfaces from point clouds," in *European Conference on Computer Vision*. Springer, 2020, pp. 108–124.

[18] M.-J. Rakotosaona, P. Guerrero, N. Aigerman, N. J. Mitra, and M. Ovsjanikov, "Learning delaunay surface elements for mesh reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 22–31.

[19] S.-L. Liu, H.-X. Guo, H. Pan, P.-S. Wang, X. Tong, and Y. Liu, "Deep implicit moving least-squares functions for 3d reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1788–1797.

[20] Z. Huang, Y. Wen, Z. Wang, J. Ren, and K. Jia, "Surface reconstruction from point clouds: A survey and a benchmark," *arXiv preprint arXiv:2205.02413*, 2022.

[21] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[22] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[23] J. Kerr, L. Fu, H. Huang, Y. Avigal, M. Tancik, J. Ichnowski, A. Kanazawa, and K. Goldberg, "Evo-nerf: Evolving nerf for sequential robot grasping of transparent objects," in *6th annual conference on robot learning*, 2022.

[24] Y. Liu, P. Wang, C. Lin, X. Long, J. Wang, L. Liu, T. Komura, and W. Wang, "Nero: Neural geometry and brdf reconstruction of reflective objects from multiview images," *arXiv preprint arXiv:2305.17398*, 2023.

[25] M. Bemana, K. Myszkowski, J. Revall Frisvad, H.-P. Seidel, and T. Ritschel, "Eikonal fields for refractive novel-view synthesis," in *ACM SIGGRAPH 2022 Conference Proceedings*, 2022, pp. 1–9.

[26] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, 2023.

[27] D. Wang, T. Zhang, and S. Süsstrunk, "Nemto: Neural environment matting for novel view and relighting synthesis of transparent objects," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 317–327.

[28] W. Deng, D. Campbell, C. Sun, S. Kanitkar, M. Shaffer, and S. Gould, "Ray deformation networks for novel view synthesis of refractive objects," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 3118–3128.

[29] A. Agrawal, R. Roy, B. P. Duisterhof, K. B. Hekkadka, H. Chen, and J. Ichnowski, "Clear-splatting: Learning residual gaussian splats for transparent object manipulation," in *RoboNerF: 1st Workshop On Neural Fields In Robotics at ICRA 2024*.

[30] M. Seeger, "Gaussian processes for machine learning," *International journal of neural systems*, vol. 14, no. 02, pp. 69–106, 2004.

[31] G. Z. Gandler, C. H. Ek, M. Björkman, R. Stolkin, and Y. Bekiroglu, "Object shape estimation and modeling, based on sparse gaussian process implicit surfaces, combining visual data and tactile exploration," *Robotics and Autonomous Systems*, vol. 126, p. 103433, 2020.

[32] S. Suresh, Z. Si, J. G. Mangelson, W. Yuan, and M. Kaess, "Shapemap 3-d: Efficient shape mapping through dense touch and vision," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7073–7080.

[33] S. Bai, J. Wang, F. Chen, and B. Englot, "Information-theoretic exploration with bayesian optimization," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1816–1822.

[34] Y. Xu, R. Zheng, S. Zhang, and M. Liu, "Bayesian generalized kernel inference for exploration of autonomous robots," *arXiv preprint arXiv:2301.00523*, 2023.

[35] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies." *Journal of Machine Learning Research*, vol. 9, no. 2, 2008.

[36] M. Deisenroth and J. W. Ng, "Distributed gaussian processes," in *International conference on machine learning*. PMLR, 2015, pp. 1481–1490.

[37] R. Zeng, Y. Wen, W. Zhao, and Y.-J. Liu, "View planning in robot active vision: A survey of systems, algorithms, and applications," *Computational Visual Media*, vol. 6, pp. 225–245, 2020.

[38] H. Dhami, V. D. Sharma, and P. Tokekar, "Pred-nbv: Prediction-guided next-best-view planning for 3d object reconstruction," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7149–7154.

[39] P. K. Murali, A. Dutta, M. Gentner, E. Burdet, R. Dahiya, and M. Kaboli, "Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4686–4693, 2022.

[40] L. Y. Chen, B. Shi, R. Lin, D. Seita, A. Ahmad, R. Cheng, T. Kollar, D. Held, and K. Goldberg, "Bagging by learning to singulate layers using interactive perception," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 3176–3183.
[41] A. Meliou, A. Krause, C. Guestrin, and J. M. Hellerstein, "Nonmyopic informative path planning in spatio-temporal models," in *AAAI*, vol. 10, no. 4, 2007, pp. 16–7.
[42] C. Wang, D. Zhu, T. Li, M. Q.-H. Meng, and C. W. De Silva, "Efficient autonomous robotic exploration with semantic road map in indoor environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2989–2996, 2019.
[43] H. Mao and J. Xiao, "Object shape estimation through touch-based continuum manipulation," in *Robotics Research: The 18th International Symposium ISRR*. Springer, 2020, pp. 573–588.
[44] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.
[45] P. Goldberg, C. Williams, and C. Bishop, "Regression with input-dependent noise: A gaussian process treatment," *Advances in neural information processing systems*, vol. 10, 1997.
[46] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.
[47] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *Seminal graphics: pioneering efforts that shaped the field*, 1998, pp. 347–353.
[48] W. F. Caselton and J. V. Zidek, "Optimal monitoring network designs," *Statistics & Probability Letters*, vol. 2, no. 4, pp. 223–227, 1984.
[49] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.
[50] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.