2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

SuPerPM: A Surgical Perception Framework Based on Deep Point Matching Learned from Physical Constrained Simulation Data

Shan Lin[1], Albert J. Miao[1], Ali Alabiad[1], Fei Liu[1], Kaiyuan Wang[1], Jingpei Lu[1], Florian Richter[1], Michael C. Yip[1], *Senior Member, IEEE*

***Abstract*— A major source of endoscopic tissue tracking errors during deformations stems from wrong data association between observed sensor measurements with previously tracked scene. To mitigate this issue, we present a surgical perception framework, SuPerPM, that leverages learning-based non-rigid point cloud matching for data association, thus accommodating larger deformations than previous approaches which relied on Iterative Closest Point (ICP) for point associations. The learning models typically require training data with ground truth point cloud correspondences, which is challenging or even impractical to collect in surgical environments. Thus, for tuning the learning model, we gather endoscopic data of soft tissue being manipulated by a surgical robot and then establish correspondences between point clouds at different time points to serve as ground truth. This was achieved by employing a position-based dynamics (PBD) simulation to ensure that the correspondences adhered to physical constraints. The proposed framework is demonstrated on several challenging surgical datasets that are characterized by large deformations, achieving superior performance over advanced surgical scene tracking algorithms.** [1]

I. Introduction

With the growing popularity of endoscopic procedures, more assistive technologies can be integrated into operating rooms. Overlaying virtual visualizations from pre-operative scans of anatomy helps surgeons identify sensitive organs during procedures [1], [2]. Further help can be done intra-operatively by identifying tissue types directly from the endoscopic image data [3], [4]. In the case of robotic surgery, automation efforts are being actively researched [5], [6]. A foundational technology for these efforts is tissue tracking and reconstruction from endoscopic images. However, this remains an unsolved challenge, with performance compromised under demanding conditions like poor lighting of the tissue, frequent occlusions, and large tissue deformations.

For endoscopic tissue tracking, the primary challenge arises from the need to establish data associations on soft tissue surfaces, which commonly lack robust features. Some methods [7], [8] use the Iterative Closest Point (ICP) algorithm [9], [10] to iteratively match nearest point pairs for estimating transformations. However, this greedy search only finds correspondences within local areas, while solving for the true correspondences would require full geometric consideration as shown in Figure 1. For rigid scenes, all points share the transformation parameters, so the estimation could converge through iteration even with such inaccurate correspondences. In contrast, in deforming scenes, each point undergoes distinct transformations, which can cause the ICP algorithm to prematurely "converge" to incorrect transformations. Alternatively, other studies have proposed to use all information in the image for data association (direct methods) [11], adopting techniques like optical flow and photometric cost [12], [13], or leveraging semantic information to offer additional guidance [13]. Yet, these approaches have not resolved the gradient-locality issue, which can still trap the estimation within a local minimum.

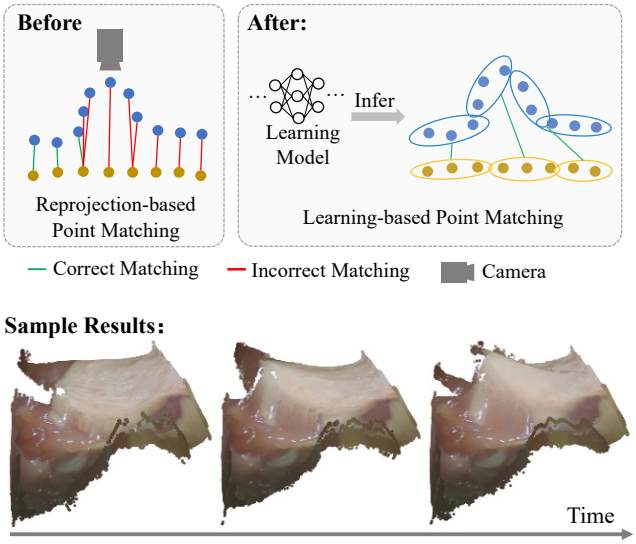

Fig. 1. We propose SuPerPM, a perception framework for endoscopic surgical scenes based on deep point matching. The perception framework, SuPer, that we built upon implements tracking based on the ICP algorithm. It projects the points onto the image plane to find the point cloud correspondences and suffers from incorrect associations (Before). We substitute the projection-based matching with a deep learning model (After). It extracts deep features, each covers the information of a certain field of view on the point cloud, and can achieve more accurate association after training. An example of tracking and reconstruction results for a deforming surgical scene is shown in the bottom of the figure.

Recently, deep learning models have demonstrated their capability to derive enhanced feature representations from 3D data [14], [15], [16], thereby enabling more robust point cloud matching [17], [18]. This paper is built upon a point-plane ICP cost-based surgical perception framework named **Su**rgical **Per**ception (SuPer) [7], [8]. We enhance the data association for the ICP cost within our framework, SuPerPM, by harnessing these advancements on learning models. Moreover, most learning-based point cloud matching models re-

[1]S. Lin, A.J. Miao, A. Alabiad, F. Liu, K. Wang, J. Lu, F. Richter, and M.C. Yip are with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA 92093, USA. (e-mail: {shl102, amiao, aalabiad, f4liu, k5wang, jil360, frichter, yip}@ucsd.edu)

[1]Our data and code are available at `https://github.com/ucsdarclab/SuPerPM.git`.

quire at least sparsely annotated correspondences between point clouds for training. However, procuring dense and accurate ground truth correspondences for surgical scenes is quite challenging. Thus, we propose a novel pipeline for generating deformed point cloud pairs for fine-tuning the point cloud matching model. This pipeline leverages the position-based dynamics (PBD) simulation [19], [20], which formulates physical constraints with positional and geometric data. PBD is capable of real-time simulations, enabling the dynamic deformation of objects. Consequently, we can establish a direct connection for mapping the physical simulation to point-cloud perception, facilitating point-wise positional mapping.

In summary, the main contributions are as follows:

- Integrate a learning-based non-rigid point cloud matching method into a surgical perception framework to improve data association for tissue tracking.
- Propose a pipeline for synthesizing non-rigid point cloud pairs using a physical constraints-based simulator (i.e., PBD), to facilitate fine-tuning of the learning-based point cloud matching method.
- Release a robotic tissue manipulation dataset with large deformations collected using the da Vinci Research Kit (dVRK) [21].
- Conduct extensive experiments on public and newly collected endoscopic data and demonstrate the performance of the proposed framework.

II. Related Work

A. Endoscopic Tissue Tracking and Reconstruction

Endoscopic tissue tracking is a specialized domain within non-rigid tracking, presenting significant challenges due to the deformable nature of the tissue. A branch of approaches relies on the ICP algorithm, which iteratively identifies nearest point pairs in the Euclidean or geodesic space for transformation estimation [12], [22], [7], [8]. However, for the deformable object, each point has a distinct transformation, and the relationships between the transformations of adjacent points are much weaker than in a rigid object. Therefore, obtaining accurate matches in early optimization iterations becomes crucial; otherwise, the transformations can quickly adapt to the wrong matches. To enhance data association, some works adopt direct methods [11] that utilize dense image information, such as photometric loss [12], [22]. Others resort to integrating additional data modalities like semantic information to guide data association [13]. Moreover, together with the aforementioned strategies, existing works typically employ regularization terms based on rigidness assumptions to mitigate the impact of noisy associations. This includes the use of the as-rigid-as-possible (ASAP) cost to ensure neighboring points move in close proximity to each other [23], [24], [25]. Yet, these regularization terms can only partially address performance degradation from incorrect data associations. In this work, we focus on further improving data association by learning-based point cloud matching.

B. Non-rigid Point Cloud Matching

Non-rigid matching between deformed point clouds is the key that influences our tissue tracking and reconstruction performance. In addition to surgical applications, precise point cloud matching is also crucial for many other tasks involving non-rigid objects. On top of ICP or photometric costs, many methods also incorporate visual features like SIFT [26] to provide additional correspondence information [27]. However, these conventional features are known to lack robustness for surgical scenes involving texture-less and moistened tissues. Recently, learning-based models have demonstrated their superior performance in representation learning for 3D data [14], [15] and identifying correspondences between data [17], [28], [29], [30], showing their advantages for non-rigid registration [31], [32]. We leverage these recent advances in learning-based non-rigid matching to provide better correspondences for tissue tracking under large deformations.

III. Method

The proposed method is developed on our previous work SuPer [7], [8]. In Section III-A, we provide a brief recap of SuPer. Then, in Section III-B, we describe how we integrate a learning-based point matching model, Lepard [17], into SuPer to enhance its data association. Surgical scenes often have flat tissue surfaces with fewer distinct features, unlike objects (e.g., animals in DeformingThings4D [33]) that can be found in many public datasets for non-rigid registration. Therefore, for fine-tuning Lepard, we present a pipeline designed to establish ground truth correspondences between deformed point cloud pairs by employing the position-based dynamics (PBD) simulation framework, ensuring the adherence to physical constraints, as outlined in Section III-C. The overview of the proposed framework is shown in Figure 2.

A. SuPer Framework

SuPer performs reconstruction and tracking of the entire scene, encompassing both the surgical tools and the deforming soft tissues. This study focuses on tissue tracking, for further insights and details on other aspects of SuPer, please see [7], [34].

1) Scene Representation: In SuPer, the tissue is tracked with a model-free method and is represented using surface elements (surfels) [35], [36]. Each surfel $\mathcal{S}$ is defined by a position $\mathbf{p}_i \in \mathbb{R}^3$, a normal $\mathbf{n}_i \in \mathbb{R}^3$, a color $\mathbf{c}_i \in \mathbb{R}^3$, a radius $\mathbb{r}_i \in \mathbb{R}$, a confidence score $\mathbb{c}_i \in \mathbb{R}$, and a timestamp $\mathbb{t}_i \in \mathbb{N}$ of when it was last updated. The quantity of surfels is directly linked to the number of image pixels, which can result in substantial computational requirements when estimating individual surfel transformations. To address this issue, SuPer employs the Embedded Deformation (ED) graph [37] that has sparser vertices (named ED nodes) to drive surfels' motions. The ED graph consists of a set of vertices $\mathcal{V}$, a set of edges $\mathcal{E}$, and a set of parameters Γ, *i.e.*, $\mathcal{G}_{ED} = \{\mathcal{V}, \mathcal{E}, \Gamma\}$. The parameters for each ED node are defined as

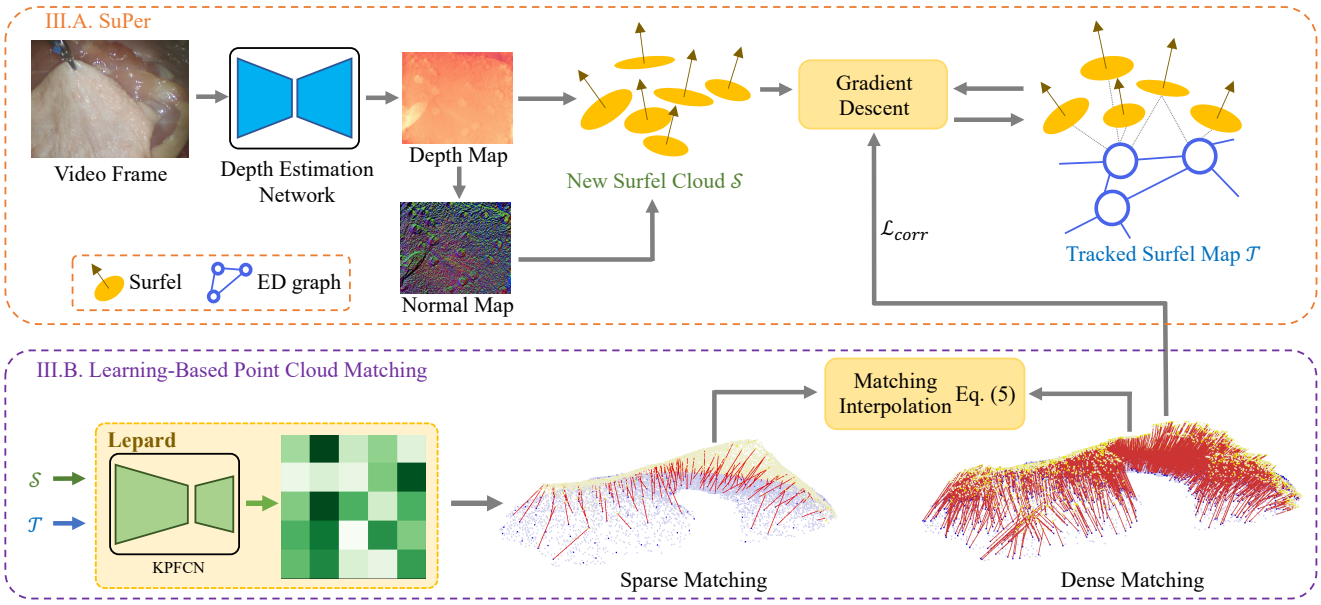

Fig. 2. Overview of SuPerPM. A learning-based point cloud matching method called Lepard [17] is integrated (Section III-B) into the surgical perception framework SuPer (Section III-A). By leveraging a data-driven method for making point-wise associations, SuPerPM is robust against poor point association that conventional approaches using ICP run into. Lepard is finetuned using synthetically generated point cloud pairs obtained using PBD simulation as described in Section III-C.

$(\mathbf{g}_j, \mathbf{q}_j, \mathbf{b}_j) \in \Gamma$, where $\mathbf{g}_j \in \mathbb{R}^3$ is the position, $\mathbf{q}_j \in \mathbb{R}^4$ and $\mathbf{b}_j \in \mathbb{R}^3$ are the quaternion and translation parameters.

2) Transformation Estimation: At every new video frame, a total of $7 \times (|\mathcal{V}| + 1)$ parameters ($|\mathcal{V}|$ is the number of ED nodes) will be estimated, *i.e.*, $\mathbf{q}_j$ and $\mathbf{b}_j$ for each ED node, and a global homogeneous transformation matrix $\mathbf{T}_g \in SE(3)$ shared by all surfels. The parameters are optimized by minimizing a total cost function (see Section III-A.3) using gradient descent [38] with PyTorch's automatic differentiation. Subsequently, the estimated parameters are utilized to update the surfel positions and normals:

$$\widetilde{\overline{\mathbf{p}}}_i = \mathbf{T}_g \sum_{j \in \mathcal{N}(\mathbf{p}_i)} \omega_j(\mathbf{p}_i)[T(\mathbf{q}_j, \mathbf{b}_j)(\overline{\mathbf{p}}_i - \vec{\mathbf{g}}_j) + \vec{\mathbf{g}}_j] \quad (1)$$

$$\widetilde{\vec{\mathbf{n}}}_i = \mathbf{T}_g \sum_{j \in \mathcal{N}(\mathbf{p}_i)} \omega_j(\mathbf{p}_i)[T(\mathbf{q}_j, 0)\vec{\mathbf{n}}_i] \quad (2)$$

where $T(\mathbf{q}_j, \mathbf{b}_j) \in SE(3)$ is the homogeneous transform matrix of the jth ED node, $\bar{\cdot}$ and $\vec{\cdot}$ are the homogeneous representations of a point and motion, *i.e.* $\overline{\mathbf{p}} = [\mathbf{p}, 1]^T$ and $\vec{\mathbf{g}} = [\mathbf{g}, 0]^T$. $\mathcal{N}(\mathbf{p}_i)$ is the set of k-nearest neighbors (KNN) of $\mathbf{p}_i$ in $\mathcal{G}_{ED}$ and is re-determined each time new positions and normals of the ED nodes and surfels are obtained. $\omega_j(\mathbf{p}_i)$ is a normalized weight that indicates the influence of $\mathbf{g}_j$ to $\mathbf{p}_i$ and is defined as $\omega_j(\mathbf{p}_i) = \frac{e^{-\|\mathbf{p}_i - \mathbf{g}_j\|}}{\sum_{j \in \mathcal{N}_i} e^{-\|\mathbf{p}_i - \mathbf{g}_j\|}}$. Equations (1) and (2) can be interpreted as that the surfels are transformed with the average motion of ED nodes near them.

3) Cost Functions: The total cost function is given by

$$\arg\min_{\mathbf{q}, \mathbf{b}, \mathbf{T}_g} \lambda_{icp}\mathcal{L}_{icp} + \lambda_r \mathcal{L}_{reg} \quad (3)$$

where $\mathcal{L}_{icp}$ is the point-to-plane ICP cost [10] that measures the similarity between the tracked data and the new observations, $\mathcal{L}_{reg}$ is the regularization term, λ_{icp} and λ_r are the hyper-parameters.

The point-to-plane ICP cost is calculated by:

$$\mathcal{L}_{icp} = \sum_i (\vec{\mathbf{n}}_o^T (\widetilde{\overline{\mathbf{p}}}_i - \overline{\mathbf{p}}_o))^2 \quad (4)$$

where $\widetilde{\overline{\mathbf{p}}}_i$ is a surfel from the tracked surfel cloud, and it is transformed using the currently estimated transformations of the ED nodes. To establish the correspondence between $\widetilde{\overline{\mathbf{p}}}_i$ and the new data, we project $\widetilde{\overline{\mathbf{p}}}_i$ onto the new image plane and conduct bilinear sampling [39] on the depth and normal maps to acquire the corresponding position and normal observations $\overline{\mathbf{p}}_o$ and $\vec{\mathbf{n}}_o$. As illustrated in Section I and II-A, during the initial iterations, these projective correspondences furnish inaccurate information that can lead the transformations toward a local minimum. In this work, we propose to mitigate this issue by replacing the ICP cost with an advanced learning-based approach to ensure more accurate associations can established from the beginning.

The regularization term is composed of two costs. One is the as-rigid-as-possible cost that enforces similar movement among neighboring ED nodes. This cost can partially mitigate the effects of incorrect data associations on the ICP algorithm. The second cost aims to ensure the estimated quaternions hold $\|\mathbf{q}\|^2 = 1$. More details and the involved equations can be found in [7].

B. Learning-based Point Cloud Matching

To enhance data association, we adapt an advanced learning-based point cloud matching method named Lepard [17]. Lepard first extracts multi-level geometric features using a convolutional backbone designed for point clouds. It then uses a transformer block to enhance these features. The transformer block consists of two layers: a self-attention layer, which aggregates global context, and a cross-attention layer, which interchanges information between the source

and target point clouds. Next, the aggregated features from the two input point clouds are compared to generate a confidence matrix $\mathcal{C}$, where each element $\mathcal{C}_{i,j}$ represents the confidence level that the corresponding points in the two point clouds are a match. Finally, matches with higher confidence values are selected as the output matches.

Specifically, Lepard takes surfel positions $\mathbf{U} \in \mathbb{R}^{N\times 3}$ of a source surfel cloud (*i.e.*, the tracked surfel cloud) and the surfel positions $\mathbf{V} \in \mathbb{R}^{M\times 3}$ of a target surfel cloud (*i.e.*, the surfel cloud extracted from the new observations) as input. $\mathbf{U}$ and $\mathbf{V}$ are first downsampled to $\mathbf{U}' \in \mathbb{R}^{N'\times 3}$ ($N' \ll N$) and $\mathbf{V}' \in \mathbb{R}^{M'\times 3}$ ($M' \ll M$) after passing through a convolutional backbone for point cloud feature extraction, resulting in a relatively sparse match set $\mathcal{K} = \{(\mathbf{u}_1, \mathbf{v}_1), (\mathbf{u}_2, \mathbf{v}_2), ..., (\mathbf{u}_K, \mathbf{v}_K)\}$, where $\mathbf{u}_k \in \mathbb{R}^3$ is a point in $\mathbf{U}'$ and $\mathbf{v}_k \in \mathbb{R}^3$ is a point in $\mathbf{V}'$. This downsampling is crucial for extracting complex, hierarchical features from the point cloud for robust matching. It is also necessary for efficient computation, as the subsequent transformer and matching blocks have a computational complexity of $O(n^2)$ [17]. However, since SuPerPM requires a dense matching between the tracked and new surfel clouds, we conduct interpolation to extend the sparse matches to dense matches. Specifically, for each surfel in the source surfel cloud, we estimate its new position $\hat{\mathbf{p}}_i$ by averaging the correspondences within the local region of its current position $\mathbf{p}_i$:

$$\hat{\mathbf{p}}_i = \mathbf{p}_i + \sum_{\mathbf{u}_j \in \mathcal{N}(\mathbf{p}_i)} \frac{(\mathbf{v}_j - \mathbf{u}_j)\|\mathbf{u}_j - \mathbf{p}_i\|^{-1}}{\sum_{\mathbf{u}_k \in \mathcal{N}(\mathbf{p}_i)} \|\mathbf{u}_k - \mathbf{p}_i\|^{-1}} \tag{5}$$

where $\mathcal{N}(\mathbf{p}_i)$ is the set of KNN of $\mathbf{p}_i$ in $\mathbf{U}'$. Then, the following point-point correspondence cost $\mathcal{L}_{corr}$ is utilized, with weight λ_c, to substitute the ICP cost $\mathcal{L}_{icp}$ in the total cost function

$$\mathcal{L}_{corr} = \sum_i \|\widetilde{\mathbf{p}}_i - \hat{\mathbf{p}}_i\|^2 \tag{6}$$

C. Deformed Point Cloud Pair Synthesis Pipeline

We have adopted the methodology introduced in [20], which utilizes position-based dynamics (PBD) for the formulation of physics-based constraints. These constraints, encompassing aspects like volume, distance, and shape matching, are crucial for ensuring stability and revealing the inherent physical properties of the system. This approach is employed to generate a sequence of simulated surface meshes, denoted as $\mathcal{M}$, each containing a set of triangle cells and vertices. Using the simulated surface meshes that is registered to the real world, our objective is to produce paired point clouds frames $\mathcal{P}_A$ and $\mathcal{P}_B$, reflecting the surface deformations induced by a physics simulation. To do so, we first project the point cloud data onto the surface mesh by finding the nearest point on $\mathcal{M}_A$ and $\mathcal{M}_B$,

$$\begin{aligned} \text{Projected } \bar{\mathcal{P}}_A &= \arg\min_{\mathbf{v}\in\mathcal{M}_A} \|\mathcal{P}_A - \mathbf{v}\| \\ \text{Projected } \bar{\mathcal{P}}_B &= \arg\min_{\mathbf{v}\in\mathcal{M}_B} \|\mathcal{P}_B - \mathbf{v}\| \end{aligned} \tag{7}$$

where $\mathbf{v}$ are located inside one of the triangle cell of the mesh. From the projected point cloud, we apply the deformation gradient tensor, $\mathbf{F}$, which is generated from the PBD simulation, and transform the points as follows

$$\begin{aligned} \bar{\mathcal{P}}_B^* &= \bar{\mathcal{P}}_A \circ \mathbf{F}^{-1} \\ \bar{\mathcal{P}}_A^* &= \bar{\mathcal{P}}_B \circ \mathbf{F} \\ \mathbf{F} &= \nabla_{\mathcal{M}_A} \mathcal{M}_B \end{aligned} \tag{8}$$

Hence, we can create the paired dataset for original point clouds $(\mathcal{P}_A \leftrightarrow \mathcal{P}_B)$ by first obtaining the indices of paired transformed points as follows:

$$\text{Paired points index} = \left\{ (\bar{\mathcal{P}}_B^*, \bar{\mathcal{P}}_B) \bigcup (\bar{\mathcal{P}}_A^*, \bar{\mathcal{P}}_A) \right\} \tag{9}$$

The whole process pipeline is show in Figure 3 It can be seen that the paired points are deformed according to the simulation with physical constraints.

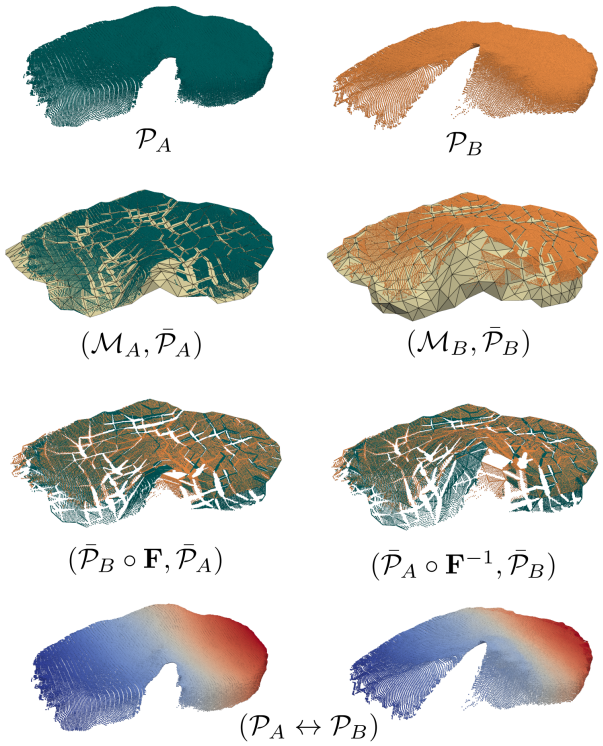

Fig. 3. Synthetic Data Generation Pipeline. We generate paired point cloud data to fine-tune Lepard [17] on tissue deformations based on real-world data that is registered to a PBD simulation. The PBD simulation ensures the associations are physically feasible.

IV. Experiments and Results

We demonstrate the proposed framework using the SuPer dataset [7], which was released alongside the original SuPer framework. Additionally, we introduce a newly collected dataset named SupDef that features larger deformations.

A. Datasets

1) SuPer: In the SuPer dataset [7], the da Vinci Research Kit (dVRK) [21], [40] was used to control a surgical robotic arm (*i.e.*, Patient Side Manipulator) to mimics the commonly performed tensioning motion in surgery by grasping and tugging a piece of chicken tissue, generating deformations of the scene. A single trial (named SuPerV1 in the following sections) that consists of 520 rectified 640×480 video frames

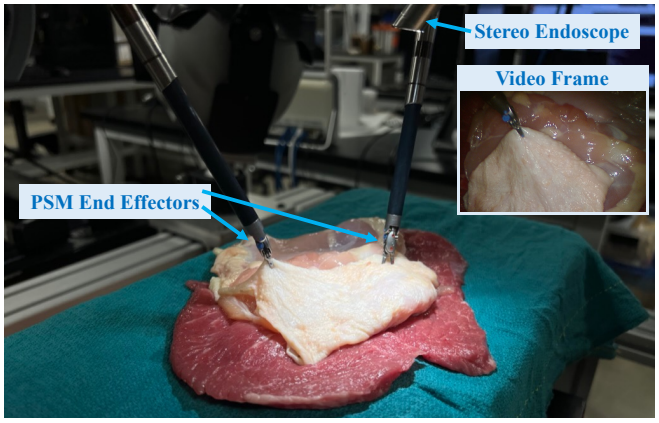

Fig. 4. Experimental Setup with the dVRK System [21]. Two PSM arms are commanded to tension and deform the tissue. Meanwhile the stereo endoscope is used to provide the image data as inputs for SuPerPM.

was annotated for evaluation. For evaluation, a total of 20 points on the tissue surfaces were chosen, and their deformation trajectories were annotated throughout the trial. We project the tracked surfels onto the image plane and compare the reprojections with their corresponding ground truth positions to calculate the reprojection errors.

2) SupDef: The deformations of tissues in existing public datasets are still fairly small. To demonstrate the benefits of SuPerPM in handling large deformations, we collected a new dataset named SupDef with relatively larger deformations across the entire manipulated tissue. The experiment setup (see Figure 4) and method for data postprocessing were identical to the SuPer dataset. The primary difference between SupDef and SuPer datasets is that the dVRK was controlled to pull the chicken tissue further away and induce larger deformations. We recorded two manipulation trials, referred to as SuperDef-T1 and SuPerDef-T2, each consisting of about 100 to 200 640×480 rectified video frames captured at 30 fps for tissue tracking. We manually annotated the trajectories of around 10∼20 selected points that undergo large deformation on the tissue surface to serve as ground truth for evaluation.

B. Evaluation Metrics

As detailed in Section IV-A, all datasets include annotated trajectories of several selected tissue surface points as ground truth. We evaluate the tracking algorithms using the average reprojection error to, defined as

$$e = \frac{1}{TS}\sum_{t=1}^{T}\sum_{s=1}^{S}\|\pi(\mathbf{p}_s^t) - \mathbf{y}_s^t\|_2 \tag{10}$$

where $\pi(\mathbf{p}_s^t)$ maps the point $\mathbf{p}_s^t$ from 3D space onto the image plane, $\mathbf{y}_s^t$ is the corresponding ground truth position of the projection of $\mathbf{p}_s^t$ in the image plane. For each trial, T is the total number of video frames and S is the total number of annotated points. We average the distance between the surfel projections and their respective ground truth positions across all annotated points and time steps in each trial.

TABLE I

REPROJECTION ERROR COMPARISON ON SUPER AND SUPDEF.

Method	Data		
	SuPerV1	SupDef-T1	SupDef-T2
DefSLAM [41]	17.1(5.5)	8.1(4.9)	**28.0(8.6)**
SD-DefSLAM [12]	27.2(18.0)	9.7(11.5)	37.9 (22.7)
SuPer [7]	9.2(13.1)	8.6(11.4)	40.7(26.7)
SuPerPM (Pre-trained)	11.1(12.3)	7.2(8.7)	43.4(27.0)
SuPerPM (Fine-tuned)	**7.9(13.1)**	**6.2(9.2)**	34.5(23.6)

* 'Pre-trained' means the Lepard model in SuPerPM is pre-trained in [17]. 'Fine-tuned' means the Lepard model in SuPerPM is fine-tuned with data generated by the proposed synthesis pipeline.
* The errors are formatted as "mean (standard deviation)". The best result in each row is in **bold**.

C. Implementation Details

To obtain the input depth maps, we employ RAFT-Stereo [42], a deep learning model designed to estimate the disparity map for rectified stereo images. We use RAFT-Stereo with its pre-trained weights, without further fine-tuning on our surgical datasets. As for tissue tracking, the procedures for initializing and adding surfels and ED nodes follow the same manner as SuPer [7]. At each new frame, we utilize the Segment Anything Model (SAM) [43] to segment tissue region from the background. Finally, we set the hyperparameters for cost functions as $\lambda_{icp} = 1$, $\lambda_r = 10$, $\lambda_c = 0.001$.

Most hyperparameters for fine-tuning Lepard follow those from Lepard's experiments on the 4DMatch Benchmark [17], except we downsample the point cloud (∼200k) to 10k points and adjust several radius values for matching and subsampling. Full details are available with the code release.

D. Results and Discussion

We compare SuPerPM to our baseline, SuPer [7], as well as advanced methods for surgical scene deformation tracking and reconstruction: DefSLAM [41] and SD-DefSLAM [12]. DefSLAM and SD-DefSLAM track scenes based on sparse feature matching and may not directly track the labeled points. To obtain the motion of a specific labeled point, we average the estimated motions of its 3 nearest neighbors.

We report the reprojection errors in Table I. SuPerPM surpasses its baseline, SuPer, upon which it is built. However, when replacing the ICP cost $\mathcal{L}_{icp}$ with the correspondence cost $\mathcal{L}_{corr}$, derived using the pre-trained Lepard model, there's a potential for performance degradation due to the large gap between the surgical data and data used to train Lepard, especially for data with larger deformations (SupDef-T2). By fine-tuning the Lepard model using data generated through the PBD-based synthesis pipeline, we significantly reduce the reprojection errors. Moreover, SuPerPM consistently outperforms SD-DefSLAM in all videos, while either matching or exceeding DefSLAM's performance. It's worth noting that both DefSLAM and SD-DefSLAM base their tracking and reconstruction on robust image features, which might result in lower reprojection errors. However, these features are usually sparse, leading to sparse reconstruction as shown in Figure 5. And in contrast to our

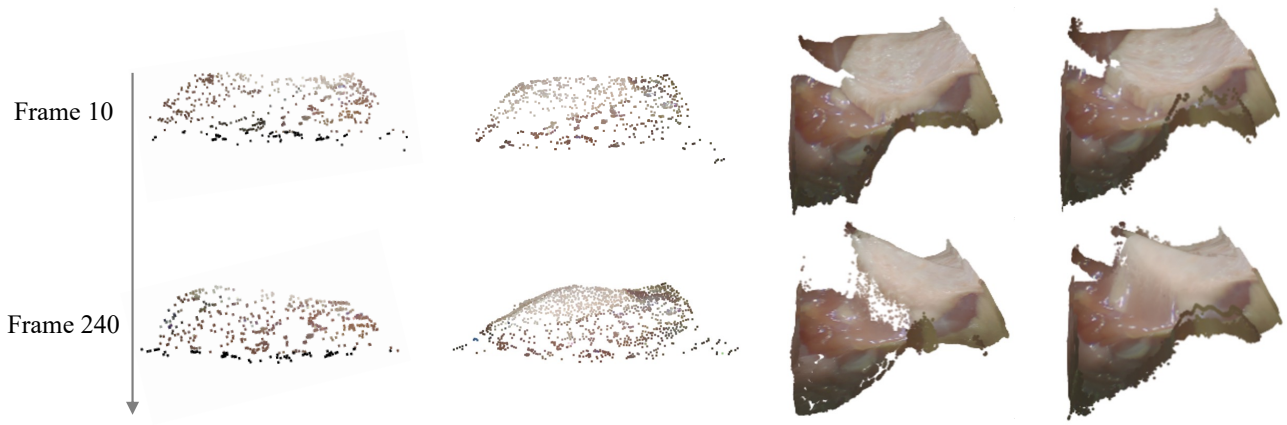

Fig. 5. Comparison of tracking results with SOTA methods.

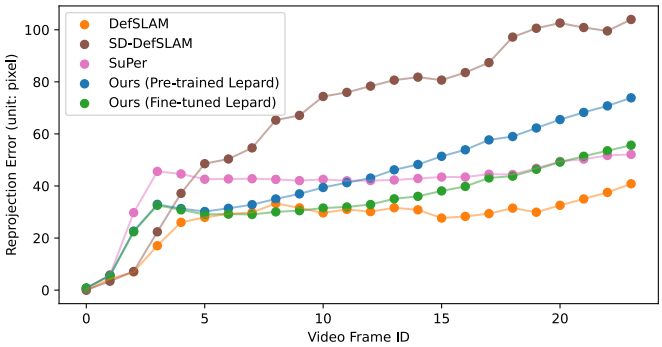

Fig. 6. Average reprojection error over time on SupDef-T2. The fine-tuned Lepard model consistently yields smaller reprojection errors compared to its pre-trained counterpart. The DefSLAM achieves the lowest reprojection errors due to only the use of sparse robust features, which is not ideal for scene reconstruction.

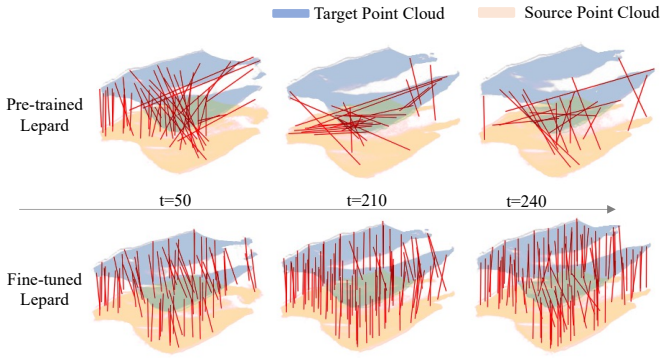

Fig. 7. Comparison of data association quality of SuPerPM across time. The pre-trained Lepard's point matching (top row) is sparse and noisy, whereas the fine-tuned Lepard (bottom row) offers denser and more consistent matches.

results, they have difficulty accurately capturing the tissue deformations caused by grasping, which are crucial for enabling autonomous tasks by the robot.

Figure 6 shows the average reprojection error at each time step in SupDef-T2, where large deformations occur throughout the entire sequence. In line with our findings from Table I, the fine-tuned Lepard model consistently yields smaller reprojection errors at every time step compared to its pre-trained counterpart. While SD-DefSLAM's performance is the worst, DefSLAM [41] achieves the lowest reprojection errors on this trail, attributing to only use sparse robust image features as mentioned above.

Furthermore, we provide examples of correspondences established by both the pre-trained and fine-tuned Lepard mode during the first optimization iteration of SuPerPM at different time steps in Figure 7. Comparing the fine-tuned Lepard model to the pre-trained Lepard model, it is evident that the fine-tuned model yields much denser and more accurate matching results. However, it is important to note that the fine-tuned model, while improved, can still produce noisy matches, as illustrated in the Figure. The errors can be attributed to two primary sources: 1) While PBD is capable of generating robust correspondences by incorporating physical constraints, it may still provide incorrect correspondences due to the inherent challenges in simulating real surgical scenes; 2) Another factor is the gap between the data used for fine-tuning and testing. Given that both error sources are inevitable to some extent, in the future, we plan to investigate techniques like RANSAC, geometric or physical constraints to identify and eliminate matching outliers for better tracking performance.

V. Conclusion

In this work, we propose a surgical perception framework SuPerPM that leverages recent advancements in deep point cloud matching. Tissue tracking approaches rely heavily on point-wise matching to update the reconstructed tissue after deformations. To achieve better point-wise matching than the previously used ICP, we use a learning-based matching model. The model is fine-tuned on synthetic data generated from PBD simulations [19], [20] of deforming tissue, hence making the model more accurate in surgical scenarios. In our current implementation, the learning model is trained separately from the surgical perception framework. Considering that our framework is built in a manner that permits gradient back-propagation, in the future, we intend to enhance the training process by training the matching model together with the optimization solver, allowing the correspondence learning to be achieved in an end-to-end manner.

REFERENCES

[1] O. Ukimura and I. S. Gill, "Imaging-assisted endoscopic surgery: Cleveland clinic experience," *Journal of endourology*, vol. 22, no. 4, pp. 803–810, 2008.

[2] F. Schulze, K. Bühler, A. Neubauer, *et al.*, "Intra-operative virtual endoscopy for image guided endonasal transsphenoidal pituitary surgery," *International journal of computer assisted radiology and surgery*, vol. 5, pp. 143–154, 2010.

[3] V. S. Prasath, "Polyp detection and segmentation from video capsule endoscopy: A review," *Journal of Imaging*, vol. 3, no. 1, p. 1, 2016.

[4] M. Grammatikopoulou, E. Flouty, A. Kadkhodamohammadi, *et al.*, "Cadis: Cataract dataset for surgical rgb-image segmentation," *Medical Image Analysis*, vol. 71, p. 102053, 2021.

[5] M. Yip and N. Das, "Robot autonomy for surgery," in *The Encyclopedia of MEDICAL ROBOTICS: Volume 1 Minimally Invasive Surgical Robotics*. World Scientific, 2019, pp. 281–313.

[6] T. Haidegger, "Autonomy for surgical robots: Concepts and paradigms," *IEEE Transactions on Medical Robotics and Bionics*, vol. 1, no. 2, pp. 65–76, 2019.

[7] Y. Li, F. Richter, J. Lu, *et al.*, "SuPer: A surgical perception framework for endoscopic tissue manipulation with surgical robotics," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2294–2301, 2020.

[8] J. Lu, A. Jayakumari, F. Richter, *et al.*, "SuPer Deep: A surgical perception framework for robotic tissue manipulation using deep learning for feature extraction," in *Proc. Int. Conf. Robot. Autom.*, pp. 4783–4789, 2021.

[9] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. Int. Conf. 3-D Digit. Imag. Model.*, pp. 145–152. IEEE, 2001.

[10] K.-L. Low, "Linear least-squares optimization for point-to-plane ICP surface registration," *Chapel Hill, University of North Carolina*, vol. 4, no. 10, pp. 1–3, 2004.

[11] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis.*, pp. 834–849. Springer, 2014.

[12] J. J. Gómez-Rodríguez, J. Lamarca, J. Morlana, *et al.*, "SD-DefSLAM: Semi-direct monocular SLAM for deformable and intracorporeal scenes," in *Proc. Int. Conf. Robot. Autom.*, pp. 5170–5177, 2021.

[13] S. Lin, A. J. Miao, J. Lu, *et al.*, "Semantic-SuPer: A semantic-aware surgical perception framework for endoscopic tissue classification, reconstruction, and tracking," in *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 4739–4746, 2023.

[14] C. Deng, O. Litany, Y. Duan, *et al.*, "Vector neurons: A general framework for SO(3)-equivariant networks," in *Proc. IEEE Int. Conf. Comput. Vision*, pp. 12 200–12 209, 2021.

[15] A. Simeonov, Y. Du, A. Tagliasacchi, *et al.*, "Neural descriptor fields: SE(3)-equivariant object representations for manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 6394–6400. IEEE, 2022.

[16] B. Thach, B. Y. Cho, A. Kuntz, *et al.*, "Learning visual shape control of novel 3D deformable objects from partial-view point clouds," in *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 8274–8281. IEEE, 2022.

[17] Y. Li and T. Harada, "Lepard: Learning partial point cloud matching in rigid and deformable scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 5554–5564, 2022.

[18] K. Fu, S. Liu, X. Luo, *et al.*, "Robust point cloud registration framework based on deep graph matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 8893–8902, 2021.

[19] J. Bender, M. Müller, and M. Macklin, "A survey on position based dynamics, 2017," in *Proceedings of the European Association for Computer Graphics: Tutorials*, ser. EG '17. Goslar, DEU: Eurographics Association, 2017. [Online]. Available: https://doi.org/10.2312/egt.20171034

[20] F. Liu, Z. Li, Y. Han, *et al.*, "Real-to-sim registration of deformable soft tissue with position-based dynamics for surgical robot autonomy," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12 328–12 334, 2021.

[21] P. Kazanzides, Z. Chen, A. Deguet, *et al.*, "An open-source research kit for the da Vinci® surgical system," in *Proc. Int. Conf. Robot. Autom.*, pp. 6434–6439, 2014.

[22] D. Recasens, J. Lamarca, J. M. Fácil, *et al.*, "Endo-depth-and-motion: Reconstruction and tracking in endoscopic videos using depth networks and photometric constraints," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7225–7232, 2021.

[23] O. G. Grasa, J. Civera, and J. Montiel, "EKF monocular SLAM with relocalization for laparoscopic sequences," in *Proc. Int. Conf. Robot. Autom.*, pp. 4816–4821, 2011.

[24] O. G. Grasa, E. Bernal, S. Casado, *et al.*, "Visual SLAM for handheld monocular endoscope," *IEEE Trans. Med. Imag.*, vol. 33, no. 1, pp. 135–146, 2013.

[25] A. Marmol, A. Banach, and T. Peynot, "Dense-ArthroSLAM: Dense intra-articular 3-D reconstruction with robust localization prior for arthroscopy," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 918–925, 2019.

[26] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.

[27] M. Innmann, M. Zollhöfer, M. Nießner, *et al.*, "VolumeDeform: Real-time volumetric non-rigid reconstruction," in *Proc. Europ. Conf. Comput. Vis.*, pp. 362–379, 2016.

[28] T. Groueix, M. Fisher, V. G. Kim, *et al.*, "3D-CODED: 3D correspondences by deep deformation," in *Proc. Eur. Conf. Comput. Vis.*, pp. 230–246, 2018.

[29] M. Niemeyer, L. Mescheder, M. Oechsle, *et al.*, "Occupancy flow: 4D reconstruction by learning particle dynamics," in *Proc. IEEE Int. Conf. Comput. Vision*, pp. 5379–5389, 2019.

[30] A. Bozic, M. Zollhofer, C. Theobalt, *et al.*, "DeepDeform: Learning non-rigid rgb-d reconstruction with semi-supervised data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 7002–7012, 2020.

[31] Y. Li, A. Bozic, T. Zhang, *et al.*, "Learning to optimize non-rigid tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 4910–4918, 2020.

[32] A. Bozic, P. Palafox, M. Zollhöfer, *et al.*, "Neural non-rigid tracking," *Advances Neural Inf. Proc. Syst.*, vol. 33, pp. 18 727–18 737, 2020.

[33] Y. Li, H. Takehara, T. Taketomi, *et al.*, "4dcomplete: Non-rigid motion estimation beyond the observable surface," in *Proc. IEEE Int. Conf. Comput. Vision*, pp. 12 706–12 716, 2021.

[34] F. Richter, J. Lu, R. K. Orosco, *et al.*, "Robotic tool tracking under partially visible kinematic chain: A unified approach," *IEEE Transactions on Robotics*, vol. 38, no. 3, pp. 1653–1670, 2022.

[35] M. Keller, D. Lefloch, M. Lambers, *et al.*, "Real-time 3D reconstruction in dynamic scenes using point-based fusion," in *Int. Conf. 3D Vision*, pp. 1–8, 2013.

[36] W. Gao and R. Tedrake, "SurfelWarp: Efficient non-volumetric single view dynamic reconstruction," *arXiv preprint arXiv:1904.13073*, 2019.

[37] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," in *ACM siggraph 2007 papers*, 2007, pp. 80–es.

[38] S. Sra, S. Nowozin, and S. J. Wright, "The tradeoffs of large scale learning," *Optimization*, pp. 351–368.

[39] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," *Advances Neural Inf. Proc. Syst.*, vol. 28, 2015.

[40] F. Richter, E. K. Funk, W. S. Park, *et al.*, "From bench to bedside: The first live robotic surgery on the dVRK to enable remote telesurgery with motion scaling," in *Int. Symp. Med. Robot.*, pp. 1–7, 2021.

[41] J. Lamarca, S. Parashar, A. Bartoli, *et al.*, "DefSLAM: Tracking and mapping of deforming scenes from monocular sequences," *IEEE Trans. Robot.*, vol. 37, no. 1, pp. 291–303, 2020.

[42] L. Lipson, Z. Teed, and J. Deng, "RAFT-Stereo: Multilevel recurrent field transforms for stereo matching," in *Int. Conf. 3D Vis.*, pp. 218–227. IEEE, 2021.

[43] A. Kirillov, E. Mintun, N. Ravi, *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.