2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Implicit Neural Field Guidance for Teleoperated Robot-assisted Surgery

Heng Zhang, Lifeng Zhu, Jiangwei Shen and Aiguo Song

Abstract— **Teleoperated techniques enable remote human-robot interaction and have been widely accepted in robot-assisted surgeries. However, it is still hard to guarantee the safety of teleoperated surgery due to the imperfect input commands limited by remote perception, preventing teleoperated surgery from being widely used. We propose a new framework to avoid the collision of surgery robots and human tissue caused by inaccurate inputs. We directly take the medical volume data and propose to use the implicit neural field to guide teleoperated robot-assisted surgery. With guidance, the trajectory of the robot manipulator is optimized to safely work inside a narrow workspace. We evaluated our method in several aspects and conducted a real-world experiment on a head phantom. Experimental results show that our proposed method can effectively avoid the collision between the surgical tool and the human tissue during teleoperation.**

I. INTRODUCTION

Over the years, researchers have been studying how to use robots to assist in surgery. Teleoperated techniques enable remote human-robot interaction and have been widely accepted in robot-assisted surgery. Limited by remote perception, teleoperated surgery still has risks if the clinician accidentally inputs incorrect or inaccurate commands, and how to ensure safety when performing robotic surgery has become a very important research direction.

Using medical volumetric data to guide robotic surgery is a popular way to help provide precise operations. For example, CT data are scanned before the surgery and used for surgical navigation. However, while robot control commonly assumes a well-modeled environment, medical images are usually noisy and with low resolution. They need to be vectorized into meshes [1] or fitted with template anatomy [2] in order to precisely guide the robotic manipulators. Because extracting a high-quality mesh from medical images is challenging, typical surgical navigation only serves the clinicians as a visualization tool in the surgery and the detailed geometry captured in medical images has not been fully exploited to directly guide the surgical robots in teleoperation. Existing works that use medical images to guide surgical robots depend on offline path planning and online registration [3], which is indirect. For online teleoperation, existing controllers that directly deal with medical images simply use primitives to approximate the lumenal region [4] and many geometric details are lost in the rough approximation.

Existing works use active constraints to assist a human to perform teleoperated surgeries. Active constraint controllers monitor the robotic manipulators with respect to the environment and are typically used to constrain the robotic manipulators within a safe region. The use of active constraints requires the proper definition of the constraints, and the constraints determine the extent of the feasible region. The constraints in previous work have been defined using rules and may result in smaller and incomplete areas, thus limiting the movement of the robotic manipulators.

In this work, we propose a new framework for collision-free navigation of robot-assisted surgery. We focus on head-and-neck surgery and aim to safely insert surgical tools through the mouth or throat with a teleoperated surgical robot. We introduce implicit neural field (INF)[36] as the representation of medical volume data. We show the power of INF to faithfully represent the distance field converted from the complex geometry hidden in the medical volume data. Furthermore, as the INF is differentiable, it is convenient to quickly query for the distance as well as its gradient. With the help of INF, we successfully guide the surgical robots with imperfect teleoperated inputs and the surgical tools are corrected to move in a collision-free way. With the proposed method, the mesh reconstruction from medical images is avoided and we ensure the safety in the teleoperation with a shared control under the implicit neural field.

Our main contributions include:

- we introduce the implicit neural field for guiding teleoperated surgical robots safely moving without mesh reconstruction,
- we validate the efficiency of INF-guided teleoperation with experiments on physical phantoms.

II. RELATED WORKS

A. Teleoperation

Teleoperation is an important technique for robot-assisted surgery [5]. Clinicians are enabled to work on remote surgery, reducing risks of cross-infection or being exposed to X-rays during surgery. For the safety of teleoperated surgery, building an immersive environment with high-quality visual and/or haptic feedback is one of the important aspects of teleoperation [6]. As erroneous commands are not avoidable in manual operations, another important solution is to have the robot intelligently correct the teleoperation inputs, avoiding undesired collisions with anatomy or surgical tools [7]. By modeling the feasible region as a set of inequalities, the robot motion can be well constrained using virtual fixtures or dynamic active constraints [8], [9]. However, it is difficult to precisely model the constrained inequalities for the feasible region bounded by freeform surfaces. If the freeform anatomy is modeled as meshes [10], [11], the

The authors are with the State Key Laboratory of Bioelectronics Jiangsu Key Lab of Remote Measurement and Control, School of Instrument Science and Engineering, Southeast University, 210096 P.R.China. Contact: lfzhulf@gmail.com

system may incorporate too many inequality constraints and is computationally prohibitive to solve. Till now, virtual fixtures in teleoperation are commonly approximated by primitives such as boxes or spheres [9]. For narrow lumenal regions, such approximation may over-constrain the robot manipulator and lead to infeasible solutions.

B. Implicit Neural Field

There are emerging works focusing on coordinate networks, which implicitly define two-dimensional (2D) or three-dimensional (3D) signals using neural networks. The input of these networks is the coordinate. They output scalar values depending on the type of application. For example, occupancy fields and signed distance fields are used to model 3D shapes. They are widely used in many applications, e.g., modeling 3D shapes[12], [13], [14], [15], [16], [17], [18], [19], [20], view synthesis[21], [22], [23], [24], [25], [26], [27], [28], [29], [30] and texture synthesis[31], [32]. These approaches model the signals or scenes using a network that can be queried at any coordinate, rather than a pixel or a grid, and provide a spatial continuity which is significant when the input data has low resolution. On the other hand, these researches most work on the fitting quality of the neural network. They only focus on the output scalars and ignore the gradient of the output value with respect to the input coordinate. We will exploit implicit neural field in combination with constrained robot manipulation for teleoperated robotic surgery.

C. Active constraints

Active constraints, also known as virtual fixtures, are widely used during teleoperated surgeries that involve human-machine collaboration to limit the movement of the robotic manipulators and provide guidance. The constraint geometries can be modeled in various ways, such as hyperplane constraints[37], polygonal mesh constraints[38], point cloud constraints[39], and neural network constraints[40]. Reconstruction of polygonal mesh for CT is challenging because the CT is complex and noisy. Active constraints can be static or dynamic. The majority of the literature uses static constraints, assuming the environment is stable enough. However, it is significant to constrain deformation during surgeries. Dynamic 3-D virtual fixtures (DVFs) were used to provide the surgeon with more helpful guidance by constraining the surgeon's hand motions thereby protecting sensitive structures[3]. The vector-field-inequalities method is proposed to provide dynamic active constraints to moving objects sharing the same workspace[4]. Our method uses an implicit neural network to represent the static constraint geometry.

III. Approach

Our goal is to directly use medical volume data to guide the teleoperation in robot-assisted surgery. In this work, we use computed tomography (CT) scans as medical images. We work on the online control of the robotic manipulator in teleoperation. The raw teleoperated motion commands mapped from the master hand are taken as the input $\{p_0, p_1, ...\}$, and they are used to query the distance and its gradient using INF. After optimizing the raw motion commands using the queried distance and gradient, we output a refined motion sequence $\{p'_0, p'_1, ...\}$ of the following hand in real-time, which avoids undesired collisions with anatomy. The pipeline of our work is illustrated in Fig. 1 and we will present the detailed steps in this section.

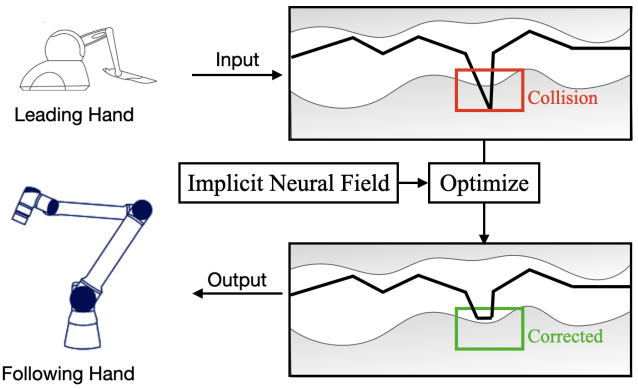


Fig. 1: The pipeline of our proposed method, the raw teleoperational motion commands from the leader hand are taken as the input. Then an optimization model is built to optimize these motion commands guided by the distance and its corresponding gradient queried from an implicit neural field. After optimization, a collision-free motion sequence will be generated in real time and sent to the following hand.

A. Implicit Neural Field

The distance field implicitly represents a shape by storing the distances to the surface in its entries. Generally, an implicit surface is represented as

$$S = \{x | F(x) = 0, \forall x \in \Omega\}, \tag{1}$$

where F is the distance field and Ω stands for the input domain of F. With a distance field, we consider the entries with zero distance on the surface. Furthermore, we are able to estimate if the point with its coordinates (x, y, z) is close to colliding with the surface by evaluating $F(x, y, z)$. Compared to explicit representations such as meshes, implicit surfaces are fast in computing the distance. However, it is difficult to find an analytic function F to represent the freeform geometry in anatomy. Therefore, F is usually stored as volume data, but this is memory-inefficient when we need a high-resolution representation.

Implicit neural field (INF) takes the advantage of neural networks to encode the complex geometry in the distance function F. It is also called a coordinate neural network which takes coordinates as the network input and produces the scalar stored at the queried position. Because of the ability of neural networks to reproduce nonlinear functions, INFs are well suited for representing curved shapes stored in the volume data. In order to use the INF to guide the teleoperated robot manipulation, we need to prepare an INF

as a surrogate model of the medical volume data. This involves an offline training phase and an online inference phase.

Data preparation. We build the INF model from medical images to guide the teleoperation. With the prepared implicit field, given the position of the surgical tool, we hope to conveniently query for its distance to the boundary of the cavity or luminal region. Therefore, we need to prepare the distance field as an input to the INF during training. As a typical class of volume data, CT scans are taken to guide the robot manipulation in our scenario. Since CT data essentially encode the density field in a 3D volume, we need to extract a distance field in the cavity region from the raw CT scans before training the guidance INF.

In the data preparation stage, we first convert the CT data into a binary volume using a simple thresholding method to mark the cavity region. In our application scenario, we consider the area filled with human tissue as a forbidden region and only allow the robot manipulator to navigate inside the natural orifice to reach the lesion. Unlike organ or bone segmentation from tissues, the density in the natural orifice is significantly different from the density of the human tissue. Therefore, it can be extracted by converting CT data into a binary volume using the simple thresholding method.

With the binary volume, we then transform it into a distance field. By applying a fast march on the binary volume from the boundary of the filled region, we are able to obtain the distance field. Due to the noise in the raw CT scans, the binary volume may also contain topological noise, which can significantly affect the result of the distance transformation. We choose to use the stochastic distance transform[34] on the discrete random sets and it has an adjustable robustness to the noise in the raw CT scans.

Training INF. In general, the neural network used in the INF is a multilayer perceptron (MLP). We illustrate the architecture of the INF in Fig 2. Note that for the sake of simplicity, we use a four-layer MLP and a 2D slice of a simple distance field to depict the INF in the figure. In our implementation, in order to improve the representational power of INF, we adopt strategies from ACORN [33], which uses a multiscale block-coordinate decomposition from an octree and a two-stage architecture to represent shapes.

During the training phase, we sample points in the space of the volume data with the coordinate (x, y, z) of each sample point and its distance d. Since the distance field is nontrivial only in the cavity region, we adaptively sample the distance field with more samples in the cavity region. The total number of epochs during training is 10000. The MSE loss is used to train the INF. We use 4 hidden layers with 512 hidden neurons. The learning rate is 0.001.

Distance inference. After training, we can directly query the distance at any given position as well as its gradient from the INF. Given a query position (x, y, z), its distance d to the nearest surface takes an evaluation of $d = F(x, y, z)$, where F is the trained neural network and a forward pass on the neural network evaluates $F(x, y, z)$. Because the neural network is fully differentiable, we can conveniently obtain

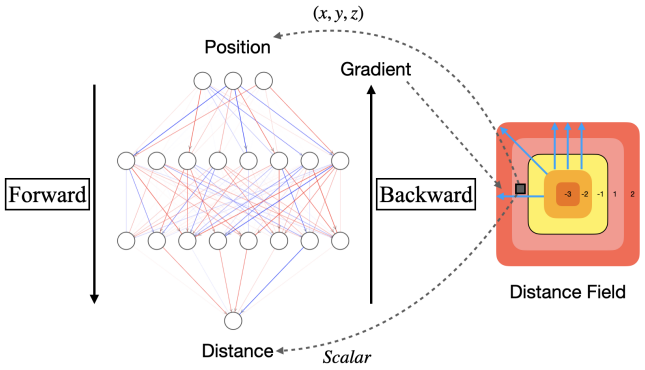


Fig. 2: The simplified architecture of INF for illustration. The INF takes the position as the input and produces the distance between the position and the nearest surface with a forward pass. Then, we generate the gradient of the distance with respect to the position with one backward pass. The distance and its gradient will be used to guide the robotic manipulation.

the gradient of the distance

$$\nabla d = \frac{\partial F(x, y, z)}{\partial (x, y, z)} \tag{2}$$

by a backward pass on the neural network F with the given input position (x, y, z). We use the PyTorch framework[41] to implement the automatic differentiation to compute the gradient.

There are two significant benefits of using an INF to represent the distance field. First, the distance can be queried directly at continuous coordinates, rather than at the discrete grid points. Because F is differentiable, it provides spatial continuity and better models the distance field of the constrained workspace in narrow regions, which is beneficial for slow and fine motion control in telesurgery. Second, the gradient of the distance can be obtained with a single backward pass. The gradient of the distance field is also continuous and suitable for robot control. Especially when the resolution of the original scan is low, the field is well moderated by INF and allows a continuous query at any point in the workspace.

B. Robot Control

We first model the surgical tool as a point to start the description of our control model. Given a point at current position p_i', and its target position p_{i+1} specified by the master hand, we have the following control model adopted from [35]

$$\begin{aligned} \arg\min_{v_i} \quad & \|v_i - (p_{i+1} - p_i')\|^2 + \lambda \|v_i\|^2 \\ \text{s.t.} \quad & v_i \cdot g_{i+1} \leq -\eta_d (d_{i+1} - d_{safe}) \end{aligned} \tag{3}$$

where $d_{i+1} = F(p_{i+1})$ and $g_{i+1} = \nabla F(p_{i+1})$ denote the distance and its gradient at the position of the next time step, η_d, λ, and d_{safe} are hyper-parameters to control the scale of the velocity, the damping factor of velocity and the safe distance to the forbidden region.

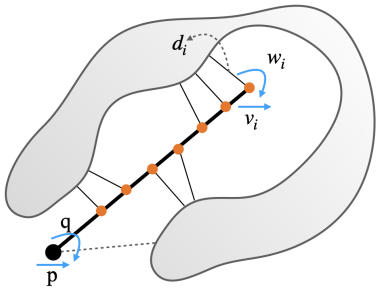


Fig. 3: The constrained motion of a stick model. We represent the pose of the stick using its position $p \in R^3$ and rotation $q \in SO(3)$ about the pivot point (shown as the black dot). We sample N points (shown as orange dots) uniformly on the stick and the gray lines denote the distances between the sampled points and the surface.

Error	Distance(mean)	Distance(max)	Gradient(mean)	Gradient(max)
Silicone Head	0.39168	1.60768	0.09638	0.37723
Real Head	0.25600	1.47978	0.076847	0.36481

TABLE I: Reconstruction errors of the INF on two models.

We further model the constrained motion of straight surgical tools such as rigid endoscopes. We model the straight surgical tool as a line segment and its radius can be modeled as the safety distance d_{safe}. As the rigid tool has 6 degrees of freedom to move, we represent the pose of the stick using its position $p \in R^3$ and rotation $q \in SO(3)$ about the pivot point. The pivot point of the stick is set as its one endpoint attached to the robot manipulator. Given the current pose(p_i', q_i') and the target pose(p_{i+1}, q_{i+1}) of the stick, we can optimize the motion of the stick using the following formula

$$\begin{aligned} \arg\min_{v_i, q_i} \quad & \|v_i - (p_{i+1} - p_i')\|^2 + \eta_q \cdot (q_{i+1} \cdot q_i'^*) \\ \text{s.t.} \quad & (v_i + w_i^{(k)}) \cdot g_{i+1}^{(k)} \leq -\eta_d (d_{i+1}^{(k)} - d_{safe}), k \in [1, N]; \end{aligned} \tag{4}$$

where we use unit quaternions to represent the rotation q_{i+1} and q_i'. We sample N points uniformly on the line segment and assume the kth sample at the pose (p_{i+1}, q_{i+1}) is $x^{(k)}$. v_i stands for the velocity of the rigid stick and all sample points share the same translational component of the velocity v_i. $w_i^{(k)}$ denotes to the velocity at x_k contributed by the rotational component of the rigid body q_i. The velocity component $w_i^{(k)}$ is calculated as $w_i^{(k)} = (x_k - p_{i+1}) \cdot w_i$, where w_i is the angular velocity of the rigid surgical tool. $d_{i+1}^{(k)}$ and $g_{i+1}^{(k)}$ are the distance to the nearest surface and its gradient of the point x_k respectively.

IV. Evaluation

In this section, we evaluated our proposed methods in different aspects. We first showed the reconstruction results of INF and then we evaluated our proposed method with a prototype of a teleoperated robot system.

A. INF capability

We tested the INF with two sets of CT data, as shown in Fig 4. In our experiment, we focused on head-and-neck surgeries and worked on the cavity and lumenal regions in human heads. As shown in the upper row, we first tested with a physical phantom made of silicone. We scanned the silicone head model to obtain its CT volume data and showed one slice of the scanned volume in Fig 4(a). The pixels in the slice stand for the density values at the corresponding coordinates. After we masked the volume data and applied the distance transform, we obtain the distance field as shown in Fig 4(b). The tissues inside the skin were not considered as we focused on the surgery path through the mouth or throat and the surgery tools would not get inside the tissue. Therefore we set the distance inside the skin to zero. Fig 4(c) showed the reconstructed INF from the distance field and the gradient field of the distance produced by INF directly. We observed that the scalar field produced by the neural network well matched with the complex nonlinear shape. Also, we highlighted the gradient of the distance in the cavity region. The gradient queried from the INF correctly implied the direction and magnitude to push the surgical tools away from the forbidden region.

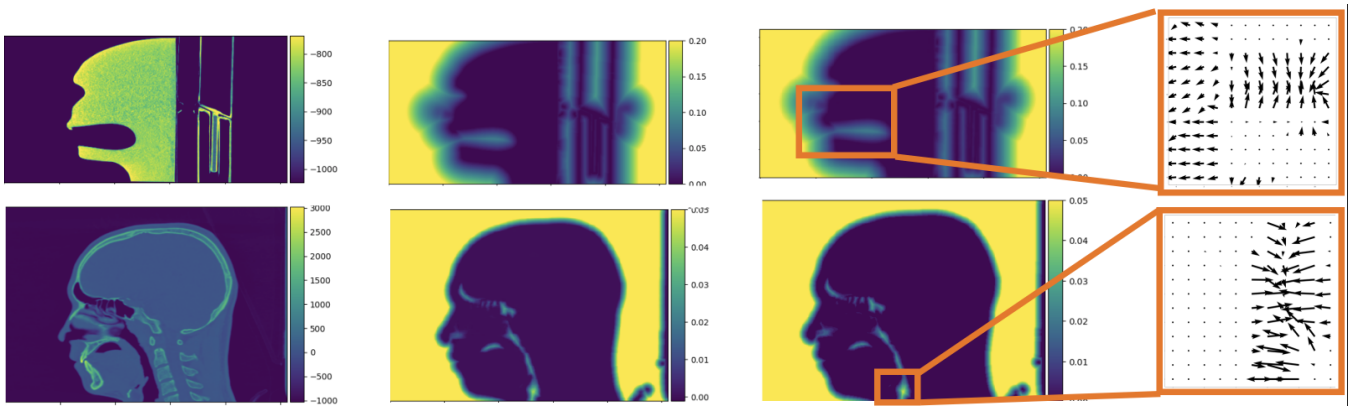
In the lower row of Fig 4, we showed the capability of INF to reconstruct CT scans with higher resolution. We took medical images scanned on a cadaveric model. The high-resolution scans revealed much detailed geometry and were different to encode in conventional mesh reconstruction. With the trained INF in our proposed method, we found the distances and the gradients inside the throat were captured by the INF accurately, making it feasible to guide the surgery tool to move inside the throat.

We quantitatively measured the reconstruction errors in our experiment and listed the statistics in Table I. The error of the distance field was computed as the pixel-wise difference between the predicted value from INF and the ground truth, where we took the pixel length as the unit in the evaluation. The error of the gradient was measured as the norm of the vector difference between the predicted gradient and the gradient extracted in the input.

We also measured the computational cost of the experiment. In our experiment, we train the INF with 6400 samples to recover the medical volume whose resolutions were $265 \times 512 \times 512$ (silicone head) and $353 \times 512 \times 512$ (real head). The offline training time was around 2 hours and the inference took around 10 ms on a graphic processing unit RTX 3090. The trained INF model representation of the volume data took disk space (61.1 MB), compared with storing the volume data directly(140.1 MB). Note that we did not optimize for the scale of the INF. We could save even more computational costs by tuning the parameters of the network without a noticeable loss of accuracy.

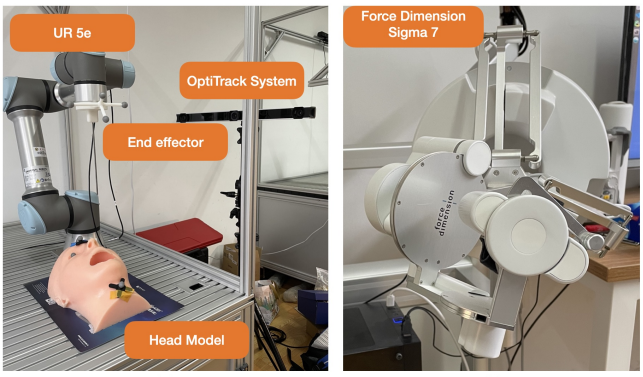
B. Teleoperation Experiment

In this subsection, we experiment with the INF-guided robot control method. We deployed our solution to a prototype teleoperated surgical robot system and set up an experiment environment as shown in Fig 5. We use a UR5e as the robot manipulator to move the end effector inside the physical phantom head. We connected a metal stick with the



(a) (b) (c)

Fig. 4: The capability of INF in representing medical volume data for teleoperated robot guidance. The upper and lower rows show the results from a scanned silicone head model and a real human head. (a) The original slice of the CT scan. (b) The distance field prepared by distance transform. (c) The distance field reconstructed by INF and the gradient of the distance field produced by INF in the cavity region.



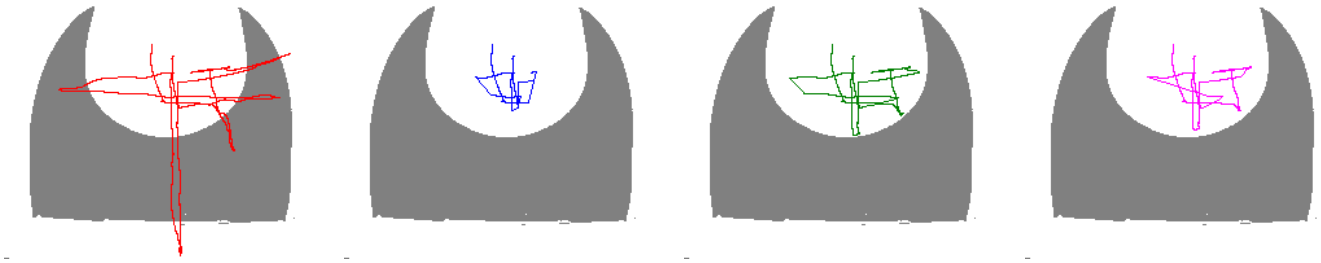
(a) (b)

Fig. 5: The experiment environment we use to evaluate our framework.

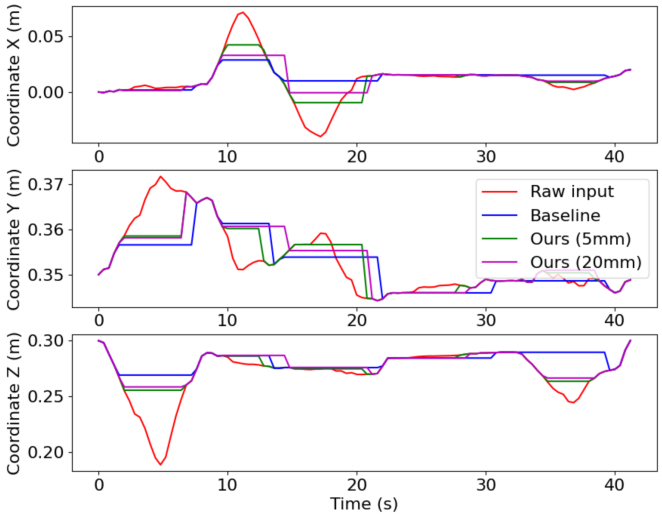
printed base on the wrist of the manipulator to simulate the surgical tool to be inserted into the head phantom. The metal stick and its base are 3D printed and their shapes are known in advance so that we could precisely control the pose of the surgical tool to the robot manipulator. In order to track the related pose between the surgical tool and the head phantom, we integrate the motion capture system OptiTrack into our experimental platform. We attached reflective markers on the base of the surgical tool as well as on the head phantom. With the world coordinate defined by the motion capturer, the poses of the surgical tool and the head phantom could be tracked in a common coordinate system for robot control. For teleoperation, the Force Dimension Sigma 7 is adopted as the leading hand, providing the input signals. We uniformly scaled the input signals to command the robot manipulator as the following hand. The CPU of the used hardware platform is Intel I5-9400F and the GPU used is NVIDIA GTX 1060.

Under the hardware settings, we used the Robot operating system (ROS) to set up the computational environment. For each time step, the system queries for the pose of the end-effector using forward kinematics and the navigation component produced the relative pose of the robot manipulator to the head model. The current relative pose and the desired pose from the leading hand were input to our control module. We used the pre-trained INF as the guidance field to query for the distance and its gradient, optimizing for a corrected velocity to safely move the manipulator inside the narrow workspace. In our implementation, the constrained optimization problem was solved by the Ceres solver. Because the dimensions of the optimization problem are small, the solver could run in real-time and it took less than 1 ms to correct the motion in each step.

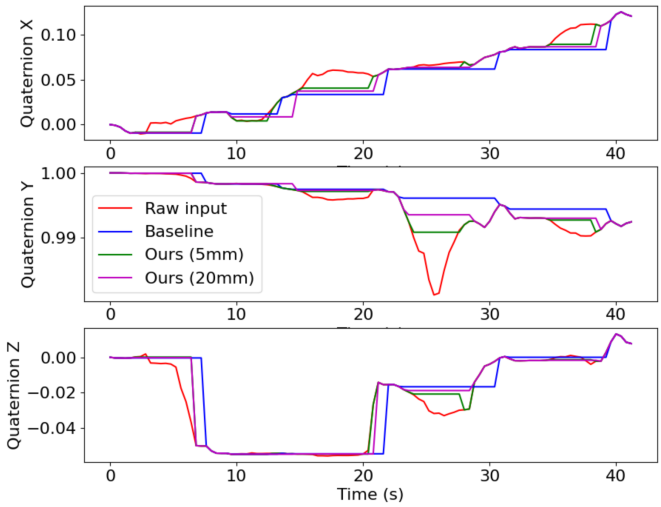
In order to evaluate our method, we compared our INF-based method with predefined constraints modeled by primitives [4] as the baseline. During our experiment, the user moved the end effector down into the mouth of the head model, moved and rotated the end effector freely, then moved upward back to the start position. During the experiment, the user kept moving and rotating the manipulator without precisely knowing whether the surgical tool collided with the mouth tissue. Taking this as the input trajectory, we recorded the output poses in our experiment. The results are illustrated in Fig 6. On a slice of the head model, we draw the projected trajectories of the surgical tool on the X-Y plane. The position and orientation of the end effector were also plotted. From the results, we found that both the baseline and our proposed method well corrected the input signals so that the end effector would not collide with the mouth tissue. Compared with our proposed method, the baseline defined the constraints using primitive shapes and had a limited approximation power to the curved cavity region, which made the robot over-constrained. In comparison, our method can reach more areas without collision. Furthermore, by adjusting the safe distance to the forbidden region, our method can define more precise constraints. As shown in Fig 6(c,d), we could flexibly adjust the safeguard distance from 5mm to 20mm, and the proposed method limited the input signals to avoid penetrating the safeguard region. The corrected signals also well followed the input trajectory when the manipulator went back to the safe area.



(a) Original (b) Baseline (c) Ours (5mm) (d) Ours (20mm)



(e) The position of the trajectory.



(f) The quaternion coefficients of the trajectory.

Fig. 6: The trajectories of the surgery tool projected on the X-Y plane (a-d) and plotted as line graphs w.r.t time (e,f).

C. Limitation and Discussion

From the experiments, we find that INF-guided teleoperation works well in our simulated surgical scenario. There are still limitations in our approach. In our experiment, it takes a few hours to fully train the INF. This is because we work to reconstruct all the medical images in the head region. If we focus on the region of interest, such as the mouth and the throat, we may have a smaller region to recover and the training time will be reduced significantly. If the training time could be reduced, we expect the INF-guide solution applicable for online CT-assisted surgeries.

In this work, we model the surgical tools in stick-like shapes. Although it models many typical rigid tools, we hope to work on more surgical tools with complex shapes for practical usage. Note that our methods rely on the sampled points on the objects while the proposed INF guidance is efficient to query for their distances and the corresponding gradients. Thus, as long as enough points can be sampled on the surgical tool and the kinematics of the surgical tools are modeled, our method applies to them without further modification. Therefore, our method may apply to other rigid bodies or even soft surgical tools. Furthermore, we assume the CT is stable and the active constraint is static. But in more practical applications, the geometries of the patients are changing over time. It requires us to model the deformation during surgeries using the implicit neural network.

V. CONCLUSIONS

To safely insert surgical tools through the mouth or throat with a teleoperated surgical robot, we propose a new framework for collision-free guidance of robot-assisted surgery based on implicit neural fields. With the distance and its gradient produced by INF, we correct the imperfect teleoperated inputs online to produce a collision-free trajectory. Our proposed method directly works on raw medical images and there is no need to extract the mesh from the complex volume data. In the future, we will apply our method to real surgery robots which can bend or deform when moving inside a narrow workspace. On the other hand, to speed up the training process, we will improve the sampling strategy during training and testing to reduce the scale of the neural work. We will also explore other acceleration methods for training the INF. Moreover, we will evaluate our method with more data and conduct animal or cadaveric studies to further improve our solution. We will also investigate the possibility of modeling the deformation of 4D-CT using an implicit neural network, to fit the needs of real teleoperated surgeries.

ACKNOWLEDGMENT

This work has been supported by the NSFC under Grants No.62133009, 92148205 and the Natural Science Foundation of Jiangsu Province under Grants No. BK20211159.

REFERENCES

[1] Chapuis, Jonas, et al. "A new system for computer-aided preoperative planning and intraoperative navigation during corrective jaw surgery." IEEE transactions on information technology in biomedicine 11.3 (2007): 274-287.

[2] Peterhans, Matthias, et al. "A navigation system for open liver surgery: design, workflow and first clinical applications." The international journal of medical robotics and computer assisted surgery 7.1 (2011): 7-16.

[3] Ren, Jing, et al. "Dynamic 3-D virtual fixtures for minimally invasive beating heart procedures." IEEE transactions on medical imaging 27.8 (2008): 1061-1070.

[4] Marinho, Murilo Marques, et al. "Dynamic active constraints for surgical robots using vector-field inequalities." IEEE Transactions on Robotics 35.5 (2019): 1166-1185.

[5] Mohan, Anmol, et al. "Telesurgery and robotics: an improved and efficient era." Cureus 13.3 (2021).

[6] Abdi, Elahe, Dana Kulić, and Elizabeth Croft. "Haptics in Teleoperated Medical Interventions: Force Measurement, Haptic Interfaces and Their Influence on User's Performance." IEEE Transactions on Biomedical Engineering 67.12 (2020): 3438-3451.

[7] Marinho, Murilo M., et al. "A unified framework for the teleoperation of surgical robots in constrained workspaces." 2019 international conference on robotics and automation (ICRA). IEEE, 2019.

[8] Funda, Janez, et al. "Constrained Cartesian motion control for teleoperated surgical robots." IEEE Transactions on Robotics and Automation 12.3 (1996): 453-465.

[9] Marinho, Murilo Marques, et al. "Dynamic active constraints for surgical robots using vector-field inequalities." IEEE Transactions on Robotics 35.5 (2019): 1166-1185.

[10] Kwok, Ka-Wai, et al. "Dynamic active constraints for hyper-redundant flexible robots." International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Berlin, Heidelberg, 2009.

[11] Li, Ming, Masaru Ishii, and Russell H. Taylor. "Spatial motion constraints using virtual fixtures generated by anatomy." IEEE Transactions on Robotics 23.1 (2007): 4-19.

[12] Chabra, Rohan, et al. "Deep local shapes: Learning local sdf priors for detailed 3d reconstruction." European Conference on Computer Vision. Springer, Cham, 2020.

[13] Chen, Zhiqin, and Hao Zhang. "Learning implicit fields for generative shape modeling." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[14] Davies, Thomas, Derek Nowrouzezahrai, and Alec Jacobson. "Overfit neural networks as a compact shape representation." (2020).

[15] Genova, Kyle, et al. "Local deep implicit functions for 3d shape." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[16] Mescheder, Lars, et al. "Occupancy networks: Learning 3d reconstruction in function space." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[17] Michalkiewicz, Mateusz, et al. "Implicit surface representations as layers in neural networks." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[18] Park, Jeong Joon, et al. "Deepsdf: Learning continuous signed distance functions for shape representation." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[19] Sitzmann, Vincent, et al. "Implicit neural representations with periodic activation functions." Advances in Neural Information Processing Systems 33 (2020): 7462-7473.

[20] Chibane, Julian, Thiemo Alldieck, and Gerard Pons-Moll. "Implicit functions in feature space for 3d shape reconstruction and completion." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[21] Chan, Eric R., et al. "pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021.

[22] Eslami, SM Ali, et al. "Neural scene representation and rendering." Science 360.6394 (2018): 1204-1210.

[23] Henzler, Philipp, Niloy J. Mitra, and Tobias Ritschel. "Escaping plato's cave: 3d shape from adversarial rendering." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[24] Jiang, Yue, et al. "Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

[25] Liu, Shaohui, et al. "Dist: Rendering deep implicit signed distance function with differentiable sphere tracing." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[26] Mildenhall, Ben, et al. "Nerf: Representing scenes as neural radiance fields for view synthesis." Communications of the ACM 65.1 (2021): 99-106.

[27] Niemeyer, Michael, et al. "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[28] Schwarz, Katja, et al. "Graf: Generative radiance fields for 3d-aware image synthesis." Advances in Neural Information Processing Systems 33 (2020): 20154-20166.

[29] Sitzmann, Vincent, Michael Zollhöfer, and Gordon Wetzstein. "Scene representation networks: Continuous 3d-structure-aware neural scene representations." Advances in Neural Information Processing Systems 32 (2019).

[30] Yariv, Lior, et al. "Multiview neural surface reconstruction by disentangling geometry and appearance." Advances in Neural Information Processing Systems 33 (2020): 2492-2502.

[31] Oechsle, Michael, et al. "Texture fields: Learning texture representations in function space." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[32] Saito, Shunsuke, et al. "Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[33] Martel, Julien NP, et al. "Acorn: adaptive coordinate networks for neural scene representation." ACM Transactions on Graphics (TOG) 40.4 (2021): 1-13.

[34] Öfverstedt, Johan, Joakim Lindblad, and Nataša Sladoje. "Stochastic distance transform: theory, algorithms and applications." Journal of Mathematical Imaging and Vision 62.5 (2020): 751-769.

[35] Marinho M M, Adorno B V, Harada K, et al. A unified framework for the teleoperation of surgical robots in constrained workspaces[C]//2019 international conference on robotics and automation (ICRA). IEEE, 2019: 2721-2727.

[36] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, Steven Lovegrove; Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 165-174

[37] S. Seung, B. Kang, W. Kim, J. Park, and S. Park, "Development of image guided master-slave system for minimal invasive brain surgery," in Proc. 41st Int. Symp. Robot. 6th German Conf. Robot., Jun. 2010, pp. 1–6.

[38] J. Aleotti, S. Caselli, and M. Reggiani, "Evaluation of virtual fixtures for a robot programming by demonstration interface," IEEE Trans. Syst., Man, Cybern. A, vol. 35, no. 4, pp. 536–545, Jul. 2005.

[39] T. Yamamoto, N. Abolhassani, S. Jung, A. M. Okamura, and T. N. Judkins, "Augmented reality and haptic interfaces for robot-assisted surgery," Int. J. Med. Robot. Comput. Assisted-Surg., vol. 8, no. 1, pp. 45–56, 2012.

[40] A.-M. Cretu, E. M. Petriu, and G. G. Patry, "Neural network architecture for 3D object representation," in Proc. IEEE Int. Workshop Haptic, Audio Visual Env. Appl., Sep. 2003, pp. 31–36.

[41] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems 32 (pp. 8024–8035).