

Robust Optical Transceiver Manipulation in Cluttered Cable Environments Using 3D Scene Understanding and Planning

Iason Sarantopoulos^{1,†}, Chenyu Liu², Bohong Weng², Sicheng Xu², Yizhong Zhang², Jiaolong Yang², Xin Tong², Fabian Otto¹, David Sweeney¹, Andromachi Chatzileftheriou¹, Antony Rowstron¹

Abstract—Robotic manipulation in cluttered environments presents significant challenges, particularly when the clutter includes thin, deformable objects like cables, which complicate perception and decision-making processes. In the context of datacenters, the automation of networking tasks often involves the manipulation of optical transceivers within densely packed cable configurations. Such environments are characterized by an abundance of delicate, overlapping, and intersecting cables, leading to frequent occlusions. This paper introduces an innovative system designed for the manipulation of optical transceivers in environments cluttered by cables. Our integrated approach combines advanced 3D scene understanding with a heuristic-based pushing policy to effectively manipulate optical transceivers amidst clutter. The system’s perception component utilizes image segmentation and 3D reconstruction to accurately model the transceivers and surrounding cables. Meanwhile, the planning aspect employs a search algorithm with task-specific heuristics, to navigate the gripper, displace obstructing cables, and safely achieve a precise pre-grasp position in front of the target transceiver. We have conducted extensive evaluations of our methodology in both simulated and real-world settings, demonstrating its high success rates, robustness, and proficiency in addressing the unique challenges posed by cable-occluded environments within datacenters.

I. INTRODUCTION

Optical fibers have become the backbone of cloud networking in modern datacenters, supporting thousands of interconnected devices and vast amounts of data traffic. The scale of networking in these facilities necessitates efficient management and maintenance to ensure optimal performance. Consequently, there is a growing need for the automation of networking operations, including the manipulation of optical transceivers and fibers. However, this automation is significantly hindered by densely packed cable arrangements.

Datacenters typically consist of numerous optical switches installed on racks, each hosting up to 96 transceivers, with many switches stacked within a single rack. This setup results in hundreds of optical cables within a rack, leading to significant cable overlap and intersection. These environments are characterized by numerous deformable and fragile optical cables routed through racks and connected to switches. The dense and complex arrangement of these cables complicates

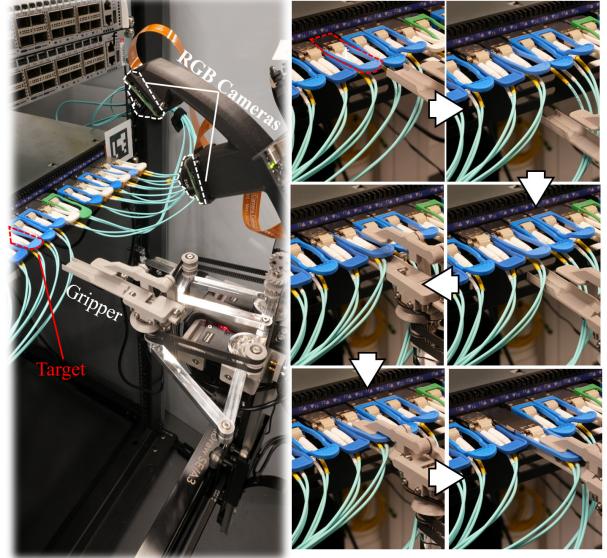


Fig. 1: Grasping and manipulating an optical transceiver in a cable-occluded environment. The gripper pushes downwards neighboring cables to reach the target pre-grasping pose and finally perform the task.

the identification and isolation of individual transceivers and makes it difficult to manipulate them without disturbing adjacent cables. The susceptibility of these cables to damage presents substantial challenges for robotic systems tasked with accurate and safe manipulation, further complicating the automation of networking operations.

Recent research has mainly focused on object manipulation in clutter involving rigid objects [1]–[7]. In contrast, the unique complexities introduced by the deformable nature of optical cables have been less frequently addressed [8]. Handling deformable objects, such as cables, ropes, and fabrics, presents significant challenges due to their complex dynamics which makes it difficult to predict the object’s behavior [9]–[11]. This unpredictability can lead to inaccurate manipulation and potential damage to the cables.

This paper addresses the challenge of dexterous manipulation in environments where occlusion arises from numerous deformable linear objects in close proximity. In such environments, there is not enough free space between the objects for the robotic gripper to be placed safely without disturbing surrounding objects. The unpredictable dynamics of deformable objects further complicates the task, as they

¹ Contributed while at Microsoft Research, Cambridge, UK

² Contributed while at Microsoft Research, Beijing, China

†Corresponding author: iason.sarantopoulos@microsoft.com

© 2025 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

make it difficult to predict behavior and isolate individual objects without disturbing adjacent ones. A typical example is the datacenter networking infrastructure, where optical fiber cables are routed in a *semi-structured* manner, and optical transceivers are placed in close proximity to each other (Fig. 1). This setting is the primary focus of this paper.

Specifically, we are interested in grasping and reseating of optical transceivers (Fig. 1) placed in ports of a switch device. The reseating process involves grasping the transceiver by its pull tab, disconnecting the transceiver from its port (without complete removal) and subsequently reconnecting it. To accomplish the task the robot should be able to identify the pull tabs and cables in the environment, plan a manipulation strategy, and navigate with minimal contact with the surrounding cables. The system should ensure safe manipulation of the optics. The density of the clutter is influenced by the degree of cable overlap and intersection, which intensifies with the number of cables and their spacing (Fig. 2). Cables are often bundled with ties, adding rigidity and increasingly occluding the ports towards the edges of the switch. This paper concentrates on environments with *moderate clutter*, featuring a single row of cables. Despite the proximity of multiple cables, some free space remains for the robot to access the target transceiver. However, the cables can be tangled and intertwined, complicating the isolation and manipulation of a single cable without affecting others. The robot must navigate around the cables with minimal contact.

To address these challenges, we propose a novel approach that combines advanced perception techniques and heuristic-based planning. Our perception module segments RGB images and reconstructs the scene in 3D to accurately model the transceivers and cables in the environment. The planning module then employs heuristic methods to find a path using A* search and navigate through the cables, pushing them aside if necessary to create sufficient space to safely grasp the transceiver. We demonstrate the effectiveness of our approach through extensive simulations and real-world experiments on a robotic platform, showcasing its robustness. While our solution is currently optimized for environments with moderate cable density, we also explore potential avenues for enhancing our approach to tackle more complex scenarios in the future. Overall, this work holds significant implications for automating networking operations in datacenters, potentially leading to increased efficiency, availability, and reliability. Beyond datacenter networking, we believe our approach has broader applications, such as in agricultural operations (e.g., fruit harvesting), where deformable linear objects (e.g. tree branches) contribute to environmental clutter.

II. RELATED WORK

a) Cable Segmentation: Learning-based methods, particularly vision foundation models, are extensively utilized for cable segmentation. [12] employs a two-stage decoder that labels and subsequently removes wires from an image, utilizing a coarse-to-fine approach. [13] utilizes the Segment Anything Model (SAM) with text prompts to segment ropes from images. For data collection, [14] captures a cable

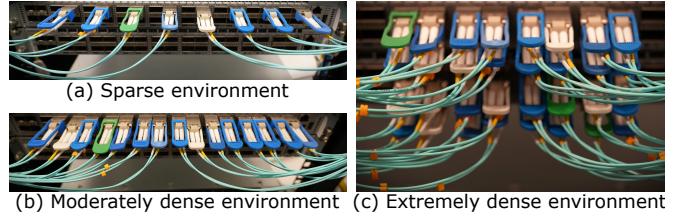


Fig. 2: The complexity of clutter depends on the degree of overlap and intersection among cables.

dataset in front of a green screen, facilitating easy background replacement. [15] proposes generating training data by overlaying synthetic foreground cables on real background images. [16] uses cable coating with UV-fluorescent material to capture cable masks under UV lighting. To the best of our knowledge, there is no publicly available dataset for cable segmentation that matches the complexity of a datacenter. Several works focus on instance segmentation to distinguish individual cables in 2D mask images. [17], [18] traces wire trajectories by grouping mask segments into super pixels. [19] segments masks into isolated parts and then connects these segments to form longer wires. [20] proposes a method to track wires by minimizing bending energy. However, all these methods necessitate a clear separation of cables in the mask images, which is often unfeasible with real-world data.

b) 3D Reconstruction: Due to the 3D nature of cables, the separation of each cable could be more efficient in 3D. However, because of the thin structure and lack of feature points, cables pose significant challenges for general-purpose multi-view stereo-based 3D reconstruction [21], [22], which often fails to produce satisfactory results. Neural Radiance Fields (NeRFs) is capable of reconstructing complex geometries with high quality, but struggles with thin wires due to the limited resolution of the grid [23]. More recently, Gaussian splatting has gained traction in 3D reconstruction [24]. Its continuous representation of 3D Gaussians makes it well-suited for representing thin wires. There have also been efforts to reconstruct wires from calibrated RGB images [25]–[27]. However, these approaches typically assume that wires are sparse in space and often fail when isolated wires cannot be clearly separated in 2D images. Additionally, there are methods for recovering curves from point clouds [28], [29], but these approaches assume the availability of a relatively complete point cloud of the cables.

c) Deformable Object Manipulation: For non-model-based deformable object manipulation, [30] selects the closest human demonstration in shape, while [10] directly learns a policy from demonstration data. In contrast, model-based approaches can utilize self-supervised learning to predict the behavior of deformable objects, with techniques that rely on image data [31], [32] and state-space information [33]. [11] introduces gradient-based action estimation to increase the complexity of tasks that can be addressed. Further, planning and control can be decoupled to accommodate different representations of deformable objects [34] or to improve error compensation in cluttered environments [35]–[37]. Sim2real

methods have also been explored [38]. Specifically focusing on cable routing, hierarchical imitation learning has been applied by [39]. [40] achieves this by using tactile feedback, which is also leveraged for cable tracking [41]. Tracking has also been further studied in scenarios with partial occlusions [42]. Notably, these approaches typically consider simpler scenarios with minimal clutter, unlike the complex environments that our work addresses.

III. PROPOSED METHODOLOGY

Our methodology comprises of two main components: perception and planning. The perception module handles the 3D reconstruction of the working environment, representing pull tabs using five key points and cables with point clouds. This reconstructed environment is then used by the planning module in order to generate and execute a path.

Our algorithms are implemented on a robot equipped with two RGB cameras mounted on a two-axis motion stage, as depicted in Fig. 1. The intrinsic and extrinsic parameters of the cameras are calibrated using a chessboard [43], and the cameras are subsequently aligned to the working space using Aruco markers attached to the machine at predetermined positions. This setup ensures knowledge of the machine's pose relative to the robot's. Additionally, we assume prior knowledge of the 3D models of the switches, which includes the port locations with respect to the switch.

A. Perception

To ensure dexterity and safe manipulation, it is crucial to perform accurate 3D reconstruction of the working environment, including the pull tabs of the optical transceivers (the target of the robot's movements) and the cables (the obstacles the robot needs to avoid). Pull tabs are used to insert and remove optical transceivers from the switch ports, vary in shape, size, and color, and may deform with use. We represent the overall shape of each pull tab using five key points, which delineate the region for the robot's gripper to hold, as shown in Fig. 3b. The cables are elongated with no fixed shape, and we use point clouds to represent them.

Since datacenters usually have strict access limitations, real data is difficult to collect. To overcome this challenge, we developed an algorithm that integrates stereo vision with segmentation. For pull tabs, key points detection is performed on segmentation images using the Segmentation Anything Model (SAM) [44]. For cables, we perform 3D reconstruction using Gaussian Splatting [24] and point cloud refinement with a fine-tuned SAM model tailored for cables. The difference between synthetic and real segmentation images is minimal, allowing us to train the neural network using synthetic segmented images, which also yields good results on real segmented images.

1) Pull Tabs: We reconstruct the geometric features of the pull tab through multi-view methods (Fig. 3). Specifically, we reconstruct each pull tab separately. The robot moves to the front of each target pull tab and captures a pair of images with the two RGB cameras, as illustrated in Fig. 3a. This multi-view setup ensures comprehensive coverage of each pull tab, enabling accurate 3D reconstruction.

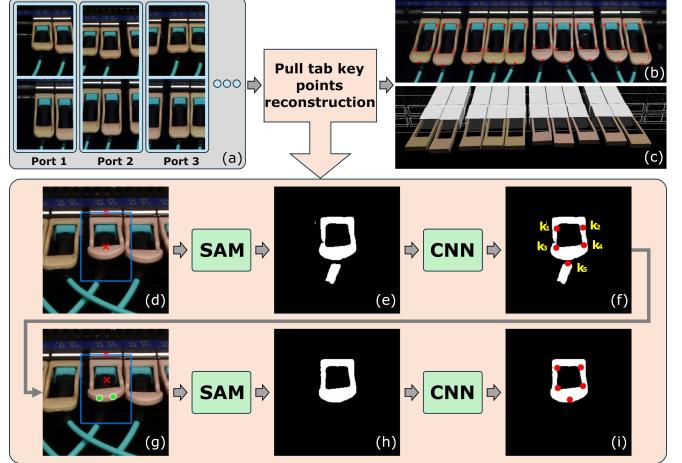


Fig. 3: Pull tab detection and reconstruction pipeline: (a) Image pairs acquired to reconstruct each pull tab. (b) Projection of reconstructed 3D key points on the image. (c) 3D simplified geometry of pull tabs. (d)-(i) Detailed steps for pull tab segmentation, key point detection and refinement.

After capturing the images, the five key points of each pull tab are detected from each image pair. The 3D coordinates of the key points are then calculated using triangulation [45] from each image pair. The projected key points of all pull tabs are shown in Fig. 3b, and the pull tabs can be visualized using simplified proxy geometry, as depicted in Fig. 3c.

To detect the key points of the pull tab from the image, SAM is employed to segment the target pull tab. The approximate position and orientation of the pull tab in the image are known due to prior calibration. As shown in Fig. 3d, the initial segmentation is achieved by providing range borders and background points as prompt to SAM. A neural network then detects five shape key points $k_1 - k_5$ (Fig. 3f) on the segmented image of the pull tab (Fig. 3e).

As the shape of pull tabs varies, the background points of the SAM prompt may be placed on the pull tab (Fig. 3d), resulting in inaccurate segmentation and key points detection. To solve this problem, we perform refinement based on the initially detected key points to improve the result. The background point is adjusted to the center of the four key points $k_1 - k_4$, while two foreground points (center of $k_3 \& k_5$ and $k_4 \& k_5$) are added to SAM prompt (Fig. 3g). This refinement ensures a clear segmentation (Fig. 3h) and more accurate key points detection (Fig. 3i).

One significant advantage of key point detection on the segmented image is that it does not require a large amount of difficult-to-obtain real data. Instead, 3D modeling software is utilized to generate extensive synthetic data for training. A 3D model of the pull tab was designed in Blender, with adjustable parameters to mimic various real-world conditions, such as bending and twisting. We generated 10,000 pull tab data samples with random parameters and trained the neural network for key points prediction using MobileNetV3 [46]. During training, we applied random cropping, noise addition, occlusion, and other enhancements to the segmented images

to improve the neural network’s robustness.

2) *Cables:* Cable reconstruction is a challenging task due to the thin structure, lack of distinctive feature points, and similar coloration among different cables. Additionally, cables are flexible and deform with robot movements, further complicating the reconstruction process. To address these challenges, we designed a two-level process: global cable reconstruction using Gaussian Splatting and multi-view local fast estimation based on segmented images.

For global cable reconstruction, the robot first scans the entire working area to capture several RGB images (e.g., over 50 positions). Utilizing the 2D Gaussian Splatting algorithm [47], we reconstruct a comprehensive point cloud of the entire scene. Concurrently, we segment the RGB images to get cables masks with cable SAM, and project the point cloud onto each mask to filter out non-cable points. Finally, we perform cable tracing to isolate each cable on the point cloud. Using Principal Component Analysis (PCA), we calculate the singular values of the point cloud in a local region ($radius < 15mm$). Points where the largest singular value (λ_1) is significantly larger than the second largest (λ_2) ($\lambda_1/\lambda_2 > 2.5$) are identified as thin tube. Then we use splines to fit and connect the tube segments, forming a coherent cable. Other points are treated as general obstacles, which are mainly cable intersection regions that are difficult to separate.

For local fast estimation, we gather several RGB images (e.g., 6 images captured at 3 positions with the two cameras) centered at the region of interest and segment these images to produce cable masks. Starting from a central cable mask, we perform ray casting [48] on each cable pixel with a step size of 1mm. Each sampling point is projected onto all the other cable masks to check whether it is projected into the cable regions. The same cable tracing method is performed on the resulting point cloud to separate each cable.

While the global reconstruction provides superior quality, it requires a few minutes to reconstruct the whole cable point cloud, making it less suitable for scenarios that demand immediate feedback. In contrast, the local reconstruction, although lower in quality, completes in seconds, making it ideal for real-time adjustments and quick situational assessments. This dual approach ensures both high-quality mapping and efficient real-time operations, leveraging the strengths of each method to achieve a balanced and adaptive perception system.

We employ a fine-tuned SAM to obtain accurate cable segmentation. Given the thin nature of cables and their extensive coverage in images, using the original SAM with point or box prompts cannot guarantee that the segmented objects are cables. We fine-tuned SAM using synthetic cable images generated in MuJoCo [49], to enhance performance. We combined those with background images of server racks for training. This fine-tuned SAM demonstrated excellent cable segmentation results on real images.

B. Planning

The planning module assumes a rectangular-shaped gripper approximated by its bounding box, with its tip’s position denoted \mathbf{p}_t . The objective is to find the shortest path from

the gripper’s initial position $\mathbf{p}_0 \in \mathbb{R}^3$ to the pre-grasping goal position $\mathbf{p}_p \in \mathbb{R}^3$, while navigating through deformable cables. The pre-grasping goal position \mathbf{p}_p is placed below the tip of the pull tab, which is known from the pull tab key point \mathbf{k}_5 given by the perception system, as shown in Fig. 5. After the gripper reaches the pre-grasping goal position, then it is guided to the final grasping position \mathbf{p}_f placed below the transceiver, given the key points $\mathbf{k}_1, \mathbf{k}_2$, to finally grasp the transceiver and reseat it.

Traditionally, motion planning assumes that every object in the scene is an obstacle to be avoided [50]. In contrast, in our task, the most ideal strategy is to clear obstacles to the pre-grasping goal position \mathbf{p}_p by contacting and pushing some cables away. We use A* search with heuristics that incorporates intuitions about the problem and guides the search towards finding paths that safely push the cables away for the gripper to reach the \mathbf{p}_p .

We model the cables with point clouds provided by the perception module, as described earlier. The planning module classifies the point clouds of the cables into two categories: target cable and obstacle cables. We call target cable the cable that it is attached to the transceiver that we want to manipulate and obstacle cables the rest of the cables. The target cable is identified by fitting a spline to each point cloud. The spline closest to \mathbf{k}_5 corresponds to the target cable. We then create two point clouds based on the point clouds provided by perception, one that includes points only from the target cable and one that includes points from every other cable. From the point clouds we disregard any points outside a neighborhood around the target to reduce the search space. This neighborhood is predefined as a box of dimensions $8 \times 4 \times 4$ cm, which means that the planner operates locally around the target, which is adequate as experiments showed in Section IV.

We consider the task successful if both of the following conditions are met:

- The gripper is positioned within a predefined distance $\epsilon = 1\text{mm}$ from the \mathbf{p}_p , i.e. $\|\mathbf{p}_t - \mathbf{p}_p\| < \epsilon$.
- After the gripper reaches \mathbf{p}_p , only the target cable is above the gripper’s bounding box.

The second condition ensures that during grasping, the target cable will be the only cable within the grasp. In any other case, closing the gripper will result to grasping an obstacle cable and potentially damage it during the manipulation of the target transceiver. Notice that the second condition can be evaluated after executing the plan in either a simulated or real environment.

A* search creates a grid of 3D points, starting from the initial position \mathbf{p}_0 and ending at the pre-grasping goal position \mathbf{p}_p , by visiting neighbors of the current position. A position will not be considered as a possible neighbor, if it:

- will move the gripper forward and will cause the gripper to collide with a cable (either the target or an obstacle). In practice, this is evaluated by checking whether a point has penetrated the front face of the gripper’s bounding box (see Fig. 5a) during the transition from its previous

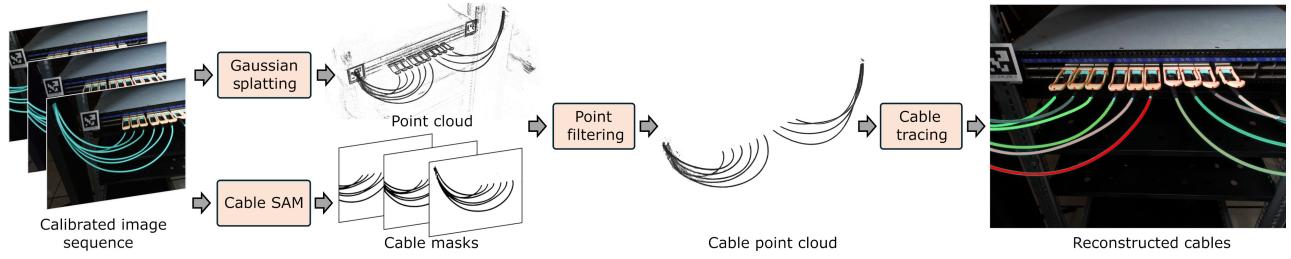


Fig. 4: Pipeline of global cable reconstruction.

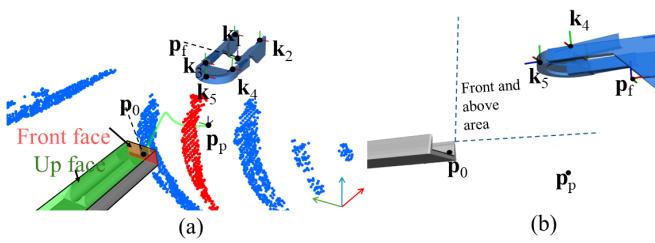


Fig. 5: Illustration of different components used in the planning module. (a) With red color the point cloud of the target cable and with blue color the point cloud of the obstacle cables. With green the final path produced by the planner. (b) Side view showing the area front and above the gripper.

position. This is necessary for the search to find a clear path towards \mathbf{p}_p .

- will move the gripper forward when there are obstacles diagonally in front and above of the gripper. This is evaluated by checking if there are obstacle points located in the area depicted in Fig. 5b. Moving forward in this case will result to placing the gripper in a position under an obstacle, and hence no progress will be made toward satisfying the second success condition.
- will move the gripper upwards and will result to a collision of the gripper to an obstacle cable. In practice, this is evaluated by checking if an obstacle point penetrated the "up face" of the gripper's bounding box (Fig. 5a) following the transition from its prior position. This constraint prevents pushing obstacles upward, which would keep them above the gripper, hindering progress toward the second success condition.

These heuristics essentially render a cable as an obstacle or not depending on the direction we approach it. Therefore, we guide the search to push the cables in certain directions that will accomplish our task. Then we use the A* algorithm to find the shortest path to \mathbf{p}_p given the above constraints. Subsequently, we interpolate the path of points from A* to a smooth trajectory that can be executed by the robot using a spline, as shown with green color in Fig. 5a. Finally, the gripper moves linearly to the final grasping position \mathbf{p}_f .

IV. EXPERIMENTAL EVALUATION

To evaluate our proposed method, we conducted a series of experiments both in simulation and in real-world conditions. The goals of the evaluation experiments are 1) to evaluate the

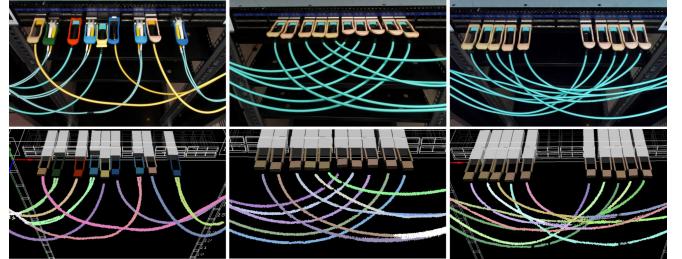


Fig. 6: A side by side comparison of a photograph of the scene (top row) and the global reconstruction result of pull tabs and cables (bottom row).



Fig. 7: Local reconstruction results of cables.

accuracy of the perception system, 2) to evaluate the accuracy of the planning system in simulation and 3) to evaluate the robustness of the integrated system in a real-world scenario.

A. Perception evaluation

To assess the robustness of our reconstruction method, we conducted experiments across a diverse set of scenarios. Fig. 6 illustrates the global reconstruction results for the entire machine. The shapes and distribution of the position of pull tabs and cables closely correspond to the actual data. Each cable is segmented and represented in a distinct color, while white regions denote densely packed areas considered as general obstacles. Fig. 7 presents the results of local reconstruction, focusing solely on cables of a local region.

Given the thin nature and shape variability of pull tabs and cables, obtaining their exact ground truth shapes is challenging. To visually evaluate the accuracy of our reconstruction, we projected the 3D key points of pull tabs and the 3D point clouds of cables onto the image of different views, examining the accuracy of aligning with real images and consistency across these projections, as depicted in Fig. 8. The pull tab key points and cable point clouds align well with the images from all viewing angles, thereby confirming the accuracy of our reconstruction. For additional results, please refer to our supplemental video.



Fig. 8: Validation of reconstruction accuracy by projecting the 3D key points of pull tabs and 3D cable point clouds onto real images from multiple views. The alignment of the projections with the real images in all views indicates a highly accurate reconstruction.

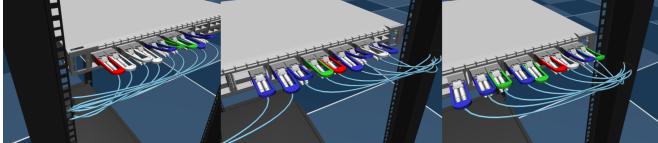


Fig. 9: Samples of simulated moderately dense datacenter scenes that we used to evaluate planning. The target transceiver is shown with a red-colored pull tab.

B. Planning evaluation in simulation

We first evaluate the method in 1000 random scenes simulated using MuJoCo [49], as shown in Fig. 9. For this purpose, we generate random simulated scenes that are mimicking the conditions of a moderately dense environment as depicted in Fig. 2b, with enough variability in the cable configuration to test the robustness of the pushing policy. First we assume we have a switch device of 36 empty ports. We randomly choose one of the 18 ports in the top row as our target port and insert a transceiver into it. Subsequently, we populate 8 ports adjacent to the target, ensuring that half of the row (9 ports) is always occupied. Each transceiver is connected to a cable, which is modeled and simulated as a cubic spline. The cable is divided into segments, each represented by capsules measuring 1 cm in length and 1 mm in radius, connected by spherical joints. In reality the cables are bundled together with ties. In simulation, we model these connection points as cylinders of radius 1.5cm and centers placed in the left and right edge of the switch. These cylinder centers are offset along the height dimension by a random value between -1 and 1 cm. For each cable, we select a point within the nearest cylinder to serve as the cable's endpoint. This approach simulates the bundling of cables, and by randomizing these endpoints, we introduce variability in cable arrangement and potential overlaps, as shown in Fig. 9.

We extract the point clouds of the cables by using as points the positions of the spherical joints the capsules are connected and feed them to the planning module. After we produce a plan and execute it in MuJoCo. A successful sample is determined by satisfying the conditions outlined in Section III-B. The overall success rate for this evaluation is 92.6%. Demonstrations of the executed plans are available in the accompanying video.

C. Real-world integrated system evaluation

Finally, we conducted experiments in a real-world setup, as shown in Fig. 1, to evaluate the robustness of our integrated system. We built a system consisting of a Cartesian gantry

motion stage and a 5-bar linkage fine positioning stage, equipped with 2 RGB cameras positioned at a 30-degree angle. The robot is tasked with reseating a transceiver inserted in a randomly selected port. We conducted 30 trials, each time changing the cable configuration of the environment. The robot captures pairs of RGB images with its two cameras, then moves to different positions to collect images for the perception module. The planner produces a path for the gripper's tip, which is executed by the robot using its inverse kinematics. The robot pushes the cables aside to create a free space for the gripper to reach the transceiver. After reaching its goal position, the robot grasps the transceiver of the target port and performs the reseating. The policy attains a success rate of 83.3%. The majority of failures occurred at the switch's edge ports, where cable density is higher due to the cable bundling to a shared endpoint. In general, the robot accurately grasped the transceivers and reinserted them without inflicting any harm to adjacent cables. Successful trials, as well as some instances of failure, are showcased in the accompanying video.

V. CONCLUSION AND FUTURE WORK

This paper addressed the challenges of dexterous manipulation in cluttered datacenter networking environments, focusing on moderately dense settings where deformable linear objects obstruct access to objects of interest. We introduced a novel method that integrates advanced perception with heuristic-based planning to effectively manipulate optical transceivers in such scenarios. This methodology was successfully implemented and validated on a real robotic platform, showcasing its robustness and efficacy in overcoming cable occlusion challenges. The findings of this study hold significant promise for automating networking operations in datacenters, potentially enhancing efficiency, availability, and reliability. Moreover, we believe that our approach has broader applications in fields such as agriculture, where environmental clutter from deformable linear objects is common.

Despite the promising results, our method has certain limitations. Color variations can affect the success of our reconstruction algorithm, particularly when segmenting black pull tabs and cables in dark environments; this could be mitigated with active lighting. Additionally, our upright camera setup limits reconstruction to upright pull tabs, making it difficult to handle transceivers positioned bottom-up or side-up. This issue could be resolved by placing the robot on a rotational stage for more versatile camera angles. Finally, to reduce the failure rates of the robot grasping, redesign of the gripper is necessary.

Our approach is currently optimized for environments with moderate cable density, featuring a single row of cables. However, the complexity of the task increases with denser environments, multiple rows of cables, and greater cable overlap and intersections. Future work will focus on extending our approach to such scenarios, exploring alternative methods for parting cables to create space for the robot to access target transceivers. We plan to construct a benchmark dataset with a high-precision 3D scanner to quantitatively

measure the reconstruction accuracy. We also plan to investigate dynamic environments where cables move as the robot interacts with them, necessitating real-time perception and planning updates. Additionally, we aim to study the impact of different camera positions on the efficiency of our approach and explore using a single camera to gather stereo information by moving the robot, reducing its form factor.

REFERENCES

- [1] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning Synergies between Pushing and Grasping with Self-supervised Deep Reinforcement Learning," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4238–4245, mar 2018. [Online]. Available: <http://arxiv.org/abs/1803.09956>
- [2] Y. Yang, H. Liang, and C. Choi, "A deep learning approach to grasping the invisible," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2232–2239, 2020.
- [3] M. R. Dogar and S. S. Srinivasa, "A planning framework for non-prehensile manipulation under clutter and uncertainty," *Autonomous Robots*, vol. 33, no. 3, pp. 217–236, 2012.
- [4] W. Bejjani, W. C. Agboh, M. R. Dogar, and M. Leonetti, "Occlusion-aware search for object retrieval in clutter," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 4678–4685.
- [5] K. Xu, H. Yu, Q. Lai, Y. Wang, and R. Xiong, "Efficient learning of goal-oriented push-grasping synergy in clutter," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6337–6344, 2021.
- [6] I. Sarantopoulos, M. Kiatos, Z. Doulgeri, and S. Malassiotis, "Split deep q-learning for robust object singulation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6225–6231.
- [7] M. Kiatos, I. Sarantopoulos, L. Koutras, S. Malassiotis, and Z. Doulgeri, "Learning push-grasping in dense clutter," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8783–8790, 2022.
- [8] O. Holešovský, R. Škoviera, and V. Hlaváč, "Movingcables: Moving cable segmentation method and dataset," *IEEE Robotics and Automation Letters*, vol. 9, no. 8, pp. 6991–6998, 2024.
- [9] N. Lv, J. Liu, and Y. Jia, "Dynamic modeling and control of deformable linear objects for single-arm and dual-arm robot manipulations," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2341–2353, 2022.
- [10] P. Sundaresan, J. Grannen, B. Thananjeyan, A. Balakrishna, M. Laskey, K. Stone, J. E. Gonzalez, and K. Goldberg, "Learning rope manipulation policies using dense object descriptors trained on synthetic depth data," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9411–9418.
- [11] A. Caporali, P. Kicki, K. Galassi, R. Zanella, K. Walas, and G. Palli, "Deformable linear objects manipulation with online model parameters estimation," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2598–2605, 2024.
- [12] M. T. Chiu, X. Zhang, Z. Wei, Y. Zhou, E. Shechtman, C. Barnes, Z. Lin, F. Kainz, S. Amirghodsi, and H. Shi, "Automatic high resolution wire segmentation and removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2183–2192.
- [13] S. Zhao, H. Zhou, L. Nanbo, L. Chen, J. Zhu, and R. B. Fisher, "A robust deformable linear object perception pipeline in 3d: From segmentation to reconstruction," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 843–850, 2023.
- [14] R. Zanella, A. Caporali, K. Tadaka, D. De Gregorio, and G. Palli, "Auto-generated wires dataset for semantic segmentation with domain-independence," in *2021 International conference on computer, control and robotics (ICCCR)*. IEEE, 2021, pp. 292–298.
- [15] R. Madaan, D. Maturana, and S. Scherer, "Wire detection using synthetic data and dilated convolutional networks for unmanned aerial vehicles," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3487–3494.
- [16] B. Thananjeyan, J. Kerr, H. Huang, J. E. Gonzalez, and K. Goldberg, "All you need is luv: Unsupervised collection of labeled images using uv-fluorescent markings," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 3241–3248.
- [17] A. Caporali, R. Zanella, D. De Greoglio, and G. Palli, "Ariadne+: Deep learning-based augmented framework for the instance segmentation of wires," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 8607–8617, 2022.
- [18] A. Caporali, K. Galassi, B. L. Žagar, R. Zanella, G. Palli, and A. C. Knoll, "Rt-dlo: Real-time deformable linear objects instance segmentation," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 11, pp. 11 333–11 342, 2023.
- [19] A. Caporali, K. Galassi, R. Zanella, and G. Palli, "Fastdlo: Fast deformable linear objects instance segmentation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9075–9082, 2022.
- [20] A. Choi, D. Tong, B. Park, D. Terzopoulos, J. Joo, and M. K. Jawed, "mbest: Realtime deformable linear object detection through minimal bending energy skeleton pixel traversals," *IEEE Robotics and Automation Letters*, 2023.
- [21] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.
- [22] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "Mvsnet: Depth inference for unstructured multi-view stereo," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 767–783.
- [23] 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," in *ECCV*, 2020.
- [24] 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, July 2023. [Online]. Available: <https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/>
- [25] L. Liu, D. Ceylan, C. Lin, W. Wang, and N. J. Mitra, "Image-based reconstruction of wire art," *ACM SIGGRAPH 2017*, 2017.
- [26] P. Wang, L. Liu, N. Chen, H.-K. Chu, C. Theobalt, and W. Wang, "Vid2curve: Simultaneous camera motion estimation and thin structure reconstruction from an rgb video," *ACM Trans. Graph. (SIGGRAPH)*, vol. 38, no. 4, 2020.
- [27] H. Choi, M. Lee, J. Kang, and D. Lee, "Online 3d edge reconstruction of wiry structures from monocular image sequences," *IEEE Robotics and Automation Letters*, 2023.
- [28] I.-K. Lee, "Curve reconstruction from unorganized points," *Computer aided geometric design*, vol. 17, no. 2, pp. 161–177, 2000.
- [29] M. Ritter, D. Schiffner, and M. Harders, "Robust reconstruction of curved line structures in noisy point clouds," *Visual informatics*, vol. 5, no. 3, pp. 1–14, 2021.
- [30] T. Tang, C. Wang, and M. Tomizuka, "A framework for manipulating deformable linear objects by coherent point drift," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3426–3433, 2018.
- [31] R. Lee, M. Hamaya, T. Murooka, Y. Ijiri, and P. Corke, "Sample-efficient learning of deformable linear object manipulation in the real world through self-supervision," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 573–580, 2022.
- [32] H. Zhang, J. Ichnowski, D. Seita, J. Wang, H. Huang, and K. Goldberg, "Robots of the lost arc: Self-supervised learning to dynamically manipulate fixed-endpoint cables," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 4560–4567.
- [33] M. Yan, Y. Zhu, N. Jin, and J. Bohg, "Self-supervised learning of state estimation for manipulating deformable linear objects," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2372–2379, 2020.
- [34] D. McConachie, A. Dobson, M. Ruan, and D. Berenson, "Manipulating deformable objects by interleaving prediction, planning, and control," *The International Journal of Robotics Research*, vol. 39, no. 8, pp. 957–982, 2020. [Online]. Available: <https://doi.org/10.1177/0278364920918299>
- [35] M. Yu, K. Lv, C. Wang, M. Tomizuka, and X. Li, "A coarse-to-fine framework for dual-arm manipulation of deformable linear objects with whole-body obstacle avoidance," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 10 153–10 159.
- [36] Y. Tang, X. Chu, J. Huang, and K. W. Samuel Au, "Learning-based mpc with safety filter for constrained deformable linear object manipulation," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2877–2884, 2024.
- [37] M. Yu, K. Lv, H. Zhong, S. Song, and X. Li, "Global model learning for large deformation control of elastic deformable linear objects: An efficient and adaptive approach," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 417–436, 2023.
- [38] D. Tong, A. Choi, L. Qin, W. Huang, J. Joo, and M. K. Jawed, "Sim2real neural controllers for physics-based robotic deployment

- of deformable linear objects,” *The International Journal of Robotics Research*, vol. 43, no. 6, pp. 791–810, 2024.
- [39] J. Luo, C. Xu, X. Geng, G. Feng, K. Fang, L. Tan, S. Schaal, and S. Levine, “Multistage cable routing through hierarchical imitation learning,” *IEEE Transactions on Robotics*, vol. 40, pp. 1476–1491, 2024.
- [40] A. Wilson, H. Jiang, W. Lian, and W. Yuan, “Cable routing and assembly using tactile-driven motion primitives,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 10408–10414.
- [41] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, “Cable manipulation with a tactile-reactive gripper,” *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1385–1401, 2021.
- [42] S. Lin, X. Jiang, and Y. Liu, “Cable manipulation with partially occluded vision feedback,” in *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2022, pp. 1245–1250.
- [43] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
- [44] A. Kirillov *et al.*, “Segment anything,” *arXiv:2304.02643*, 2023.
- [45] R. I. Hartley and P. Sturm, “Triangulation,” *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146–157, 1997.
- [46] A. Howard *et al.*, “Searching for mobilenetv3,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1314–1324.
- [47] B. Huang, Z. Yu, A. Chen, A. Geiger, and S. Gao, “2d gaussian splatting for geometrically accurate radiance fields,” in *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024.
- [48] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, “Kinectfusion: Real-time dense surface mapping and tracking,” in *2011 10th IEEE international symposium on mixed and augmented reality*. Ieee, 2011, pp. 127–136.
- [49] E. Todorov, T. Erez, and Y. Tassa, “MuJoCo: A physics engine for model-based control,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE. IEEE, oct 2012, pp. 5026–5033. [Online]. Available: <http://ieeexplore.ieee.org/document/6386109/>
- [50] K. Lynch, *Modern Robotics*. Cambridge University Press, 2017.