HANDLOOM 3.0: Interactive Bi-Directional Cable Tracing Amid Clutter

Justin Yu[*1], Nidhya Shivakumar[*1], Veena Sumedh[*1], Josh Zhang[1], Ethan Ransing[1], Osher Azulay[1], Ken Goldberg[1]

Abstract— **Accurate state estimation of deformable cables is crucial in robotic perception, with significant applications in industrial automation and surgical robotics. However, resolving the state of multiple cluttered and tangled cable configurations poses challenges due to occlusions, overlapping cables, and ambiguous crossings. We introduce HANDLOOM 3.0, which combines bidirectional cable tracing with novel interactive perception primitives—Divergence Push and Cluster Dilation—to actively resolve ambiguities caused by occlusions, crossings, and dense cable arrangements. HANDLOOM 3.0 selects intervention primitives based on uncertainty in state estimates. Extensive evaluations on physical scenarios suggest that HANDLOOM 3.0 achieves on average 25.9% improvement in the percentage of cables correctly traced over prior methods. Project website: `https://nidhya-s.github.io/handloom3.0/`.**

I. Introduction

The manipulation and perception of deformable linear objects (DLOs) such as cables, hoses, and threads present unique challenges due to their complex deformations, self-occlusions, and entanglements. In industrial and assembly settings, cluttered cables reduce safety and complicate troubleshooting, requiring efficient organization and state estimation.

Single-cable tracing methods that use geometric models [1] and data-driven techniques [2] have shown promise. HANDLOOM [3] introduced an RGB vision-based framework that sequentially traces cables, but struggles in multi-cable scenarios with overlaps and ambiguous crossings [4].

Interactive Perception (IP) addresses these ambiguities by actively manipulating objects to reveal occluded states [5]. The MANIP framework [4] and HANDLOOM 2.0 extended IP to multi-cable scenarios, improving tracing through modular manipulation strategies. However, HANDLOOM 2.0 lacked global consistency analysis and relied on local heuristics.

In this work, we propose HANDLOOM 3.0, a framework for reconstructing multiple cable trajectories in cluttered semi-planar environments. HANDLOOM 3.0 introduces bidirectional tracing and targeted interactive perception primitives to resolve ambiguities. Unlike prior methods, HANDLOOM 3.0 identifies intervention points from conflicting state estimates and leverages both local and global geometry to guide interventions.

Our contributions are:

[1] AUTOLab at the University of California, Berkeley
[*] Equal contribution

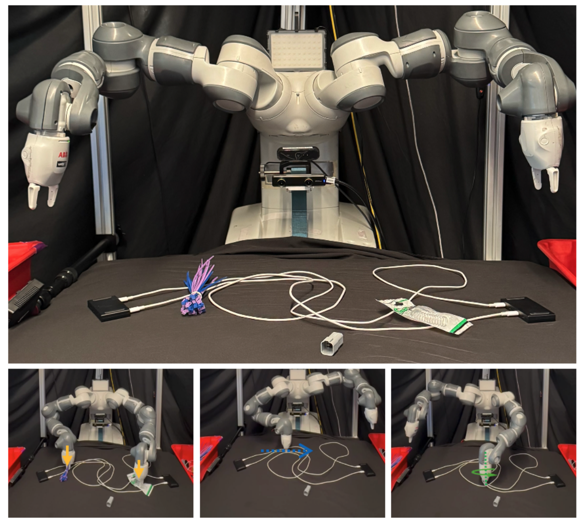

Fig. 1: Overview of the HANDLOOM 3.0 framework. Top: cluttered scene with cables and objects. Bottom: three interactive perception primitives—Object Decluttering, Divergence Push, and Cluster Dilation.

1) The HANDLOOM 3.0 framework for multi-cable tracing with divergence point detection based on integrated state estimates.
2) Two task-oriented IP primitives—Divergence Push and Cluster Dilation—combining geometric cues and Hessian filtering.
3) Real-world evaluation on 110 cluttered scenes, demonstrating significant improvements over HANDLOOM 2.0.

II. Related Work

Early cable tracing approaches relied on analytical methods, optimizing spline continuity [6–9], and visual feature tracking [10, 11]. However, these methods often fail in real-world settings with dense, occluded, or tangled cables due to strong geometric assumptions. Learning-based methods have since emerged, which use instance segmentation to predict cable structures [10, 12]. RT-DLO [13] extracts skeletons from semantic masks, while [14] uses iterative refinement. These methods struggle with long cables and complex crossings.

HANDLOOM [3] introduced a learning-based, autoregressive tracing framework using a UNet to predict trace points from cropped image patches. Though effective in structured scenes, HANDLOOM fails under heavy clutter, occlusions, and high-density crossings typical of industrial environments.

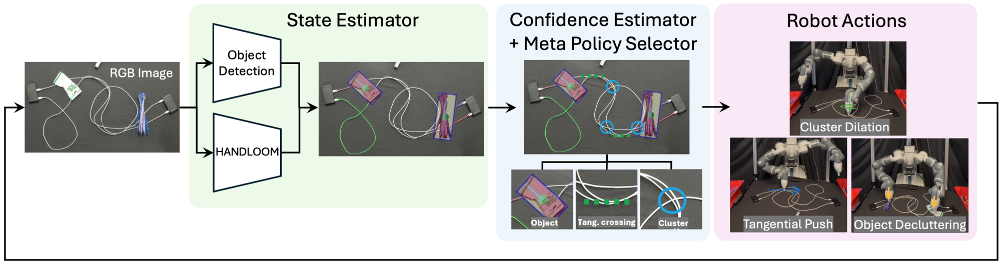

Fig. 2: Overview of the HANDLOOM 3.0 architecture. Occlusions trigger object removal. Divergence points are identified and classified into tangential crossings or cable clusters, prompting IP actions.

Interactive Perception (IP) addresses such ambiguities by actively manipulating the scene [5]. SGTM 2.0 [2] combined analytical tracing with manipulation primitives to improve accuracy. MANIP [4] extended this approach by integrating HANDLOOM with manipulation policies, introducing multi-cable interventions. HANDLOOM 2.0 demonstrated improved tracing through local uncertainty-based interventions but lacked mechanisms for global consistency.

This work advances prior efforts by handling denser scenes with object occlusions and frequent crossings. We refine intervention strategies through bi-directional tracing and analysis of global and local geometry, improving robustness in complex multi-cable configurations.

III. Method

We propose HANDLOOM 3.0, an interactive framework for reconstructing multiple deformable cable trajectories in cluttered semi-planar environments. The system iteratively refines visual estimates using bi-directional tracing and targeted robot interactions, addressing challenges from occlusions, tangles, and ambiguous crossings (Figure 2).

A. Problem Formulation

Given an overhead RGB image $\mathbf{I_t} \in \mathcal{I}$, the goal is to reconstruct the complete cable state $\hat{\mathcal{C}}_t = \bigcup_{i=1}^{n} \hat{C}_{i,t}$, where each cable trajectory $\hat{C}_{i,t}$ preserves continuity and endpoint connectivity. Each cable i follows a trajectory $\theta_i(s) = \mathbf{x}(s)$ for $s \in [0, 1]$ in workspace $\mathcal{W} \subset \mathbb{R}^3$. We assume (1) cables are visually separable from a monochrome background, (2) clutter objects are graspable, and (3) cable endpoints are fixed in terminals.

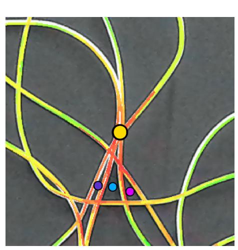

Fig. 3: **Cable Density & Cluster Dilation** The cables are overlaid with a density heatmap, where red indicates high local cable density. The central (yellow) dot marks the divergence point classified as a cluster due to the high-density region. The three dots below represent the centroids of candidate open regions, with the center dot corresponding to the open region with the largest pixel area selected for the Cluster Dilation action.

B. Pipeline Overview

As illustrated in Figure 2, HANDLOOM 3.0 first detects cable endpoints using a trained Faster R-CNN model [15]. HANDLOOM tracing is initiated bi-directionally from each endpoint, producing forward and backward trajectory estimates. Divergence points are identified where traces fail to connect or overlap incorrectly, signaling ambiguities.

When visual ambiguity is detected, the system applies targeted interactive perception (IP) primitives: Object Decluttering, Divergence Push, or Cluster Dilation. Each primitive is designed to reduce uncertainty by actively modifying the workspace state $\mathcal{W}_{t+1}$. This process iterates until the reconstructed cable set $\hat{\mathcal{C}}$ is topologically consistent.

C. Endpoint Detection and Cable Tracing

The endpoint detector outputs $\mathcal{E}(\mathbf{I_t}) = \{e_1, e_2, \ldots, e_n\}$, where $e_i \in \mathbb{R}^2$. Each endpoint initializes the HANDLOOM tracer [3], which autoregressively predicts the next trace point based on visual features. Tracing terminates upon reaching another endpoint, forming loops, exceeding time limits, or detecting occlusions.

D. Bi-Directional Tracing and Divergence Analysis

Each cable is traced in both directions. Divergence points are detected when traces from opposing endpoints fail to connect or overlap ambiguously. Divergence points are classified based on local cable density $\rho(d_i)$: clusters ($\rho(d_i) \geq \tau_c$) or tangential crossings ($\rho(d_i) < \tau_c$). This classification informs subsequent IP actions.

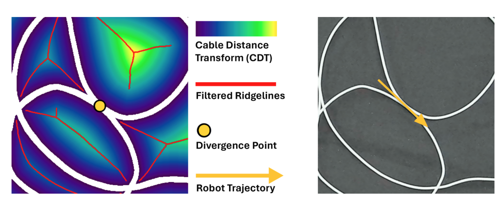

Fig. 4: (Left) CDT and ridge lines. Divergence points (yellow) at crossings are resolved with push actions along ridges. (Right) Divergence Push trajectory.

E. Interactive Perception Primitives

Object Decluttering: External objects obstructing predicted cable traces are identified using DETIC [16]. The robot executes pick-and-place actions using oriented bounding boxes, utilizing both arms for parallel removal.

Cluster Dilation: Clusters are addressed by identifying nearby open regions in the Cable Distance Transform (CDT)

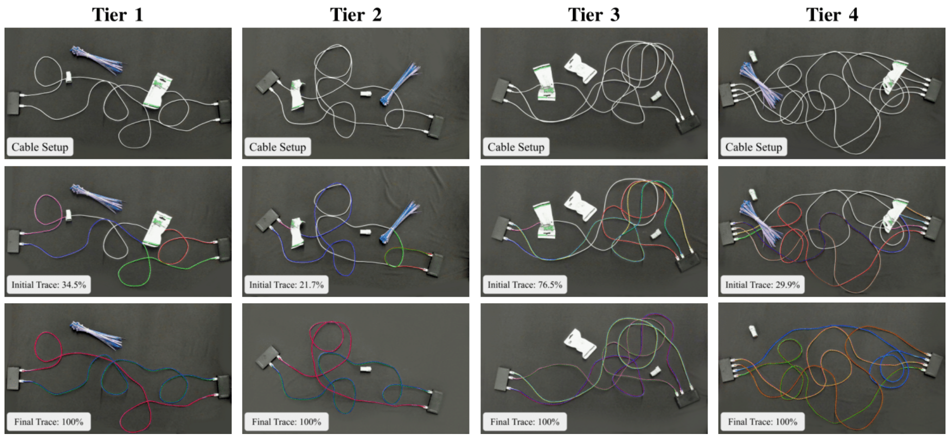

Fig. 5: Evaluation tiers. First row: initial cluttered scene; second row: initial traces; third row: completed traces after IP actions.

TABLE I: Tier System for Evaluation

	Tier 1	Tier 2	Tier 3	Tier 4
# Cables	2	2	3	4
# Tangential Points	2	3	3-4	4-5
# Objects	3-4	3-4	3-4	3-4

$D_c(p) = \min_{q \in C} \|p - q\|$. A controlled gripper opening action is performed in the largest open region (Figure 3).

Divergence Push: For tangential crossings, CDT ridges are extracted via Frangi vesselness filtering. Push actions follow these ridges to physically separate ambiguous traces (Figure 4).

IV. Experiments

A. Setup

Experiments were conducted using the ABB YuMi robot and overhead RGB camera setup shown in Figure 1. Up to four white 6-foot USB-C cables and eight clutter objects (blocks, zip ties, sockets, bags) were randomly arranged (Appendix).

B. Evaluation Protocol

We evaluate HANDLOOM 3.0 across four tiers of increasing complexity, based on cable count, tangential crossings, and clutter (Table I). Figure 5 shows examples.

C. Multi-Cable Tracing Results

Table II reports trace completion success across 50 trials without clutter. HANDLOOM 3.0 consistently outperforms HANDLOOM 2.0, especially in higher tiers with dense crossings.

TABLE II: Success rates (50 trials without clutter)

Method	Tier 1	Tier 2	Tier 3	Tier 4
HANDLOOM 2.0	89.5%	86.2%	60.0%	68.2%
HANDLOOM 3.0	**99.1%**	**91.2%**	**94.2%**	**89.4%**
Improvement	10.7%	5.8%	56.9%	31.1%

D. Tracing with Clutter

We evaluated HANDLOOM 3.0 on 60 trials with object interference (Table III). Despite clutter, the system maintained high success rates, particularly in lower tiers.

TABLE III: Success rates (60 trials with clutter)

Method	Tier 1	Tier 2	Tier 3	Tier 4
HANDLOOM 3.0	94.4%	91.4%	88.3%	82.9%

TABLE IV: Distribution of IP primitives

Primitive	Tier 1	Tier 2	Tier 3	Tier 4
Total IP Actions	3.25	4.40	6.93	7.87
Decluttering (%)	42.9%	43.4%	20.6%	19.2%
Cluster Dilation (%)	6.3%	12.1%	15.9%	31.3%
Divergence Push (%)	50.8%	44.4%	63.5%	49.4%

E. Primitive Usage Analysis

Table IV summarizes IP primitive usage across tiers. As scene complexity increases, Cluster Dilation actions become more frequent, reflecting higher cable density.

V. Limitations

HANDLOOM 3.0 inherits assumptions from HANDLOOM 2.0, requiring visually distinguishable cables in semi-planar arrangements, with most of the segments visible from above. Although HANDLOOM 3.0 adds interactive perception to resolve occlusions, HANDLOOM 3.0 still depends on a consistent cable appearance. Significant variations in thickness, elasticity, or reflectance could degrade heatmap predictions.

A monocular overhead camera further limits depth perception, making it difficult to distinguish closely spaced cables that overlap in 2D. Although bidirectional tracing and interactive perception mitigate this, scenes with many thin cables stacked vertically remain challenging.

VI. Future Work

HANDLOOM 3.0's underlying principles of visual tracking, bi-directional tracking, and active interventions have potential for extension to broader challenges in manipulating deformable objects. In particular, these techniques could support cable routing tasks through the active planning and repositioning of cables into desired, organized layouts.

APPENDIX

Fig. 6: Objects used to simulate occlusions in cluttered environments.

REFERENCES

[1] V. Viswanath *et al.*, “Autonomously Untangling Long Cables,” in *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, Jun. 2022.

[2] K. Shivakumar *et al.*, “Sgtm 2.0: Autonomously untangling long cables using interactive perception,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023, pp. 5837–5843.

[3] V. Viswanath *et al.*, “Handloom: Learned tracing of one-dimensional objects for inspection and manipulation,” in *Conference on Robot Learning*, PMLR, 2023, pp. 341–357.

[4] J. Yu *et al.*, “Manip: A modular architecture for integrating interactive perception for robot manipulation,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2024, pp. 1283–1289.

[5] J. Bohg *et al.*, “Interactive perception: Leveraging action in perception and perception in action,” *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1273–1291, 2017.

[6] A. Keipour, M. Bandari, and S. Schaal, “Deformable one-dimensional object detection for routing and manipulation,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4329–4336, 2022.

[7] P. Kicki, A. Szymko, and K. Walas, “Dloftbs–fast tracking of deformable linear objects with b-splines,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023, pp. 7104–7110.

[8] X. Huang, D. Chen, Y. Guo, X. Jiang, and Y. Liu, “Untangling multiple deformable linear objects in unknown quantities with complex backgrounds,” *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 1, pp. 671–683, 2024.

[9] 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*, vol. 8, no. 8, pp. 4863–4870, 2023.

[10] W. H. Lui and A. Saxena, “Tangled: Learning to untangle ropes with rgb-d perception,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 837–844.

[11] T. Tang, Y. Fan, H.-C. Lin, and M. Tomizuka, “State estimation for deformable objects by point registration and dynamic simulation,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2017, pp. 2427–2433.

[12] 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.

[13] 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.

[14] 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.

[15] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[16] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra, “Detecting twenty-thousand classes using image-level supervision,” in *European conference on computer vision*, Springer, 2022, pp. 350–368.