Multi-View Model-Based Visual Tracking of Deformable Linear Objects

Alessio Caporali, Gianluca Palli

***Abstract*—This paper addresses the challenge of tracking the state of DLOs during robotic manipulation, a key requirement for achieving accurate and reliable control in industrial scenarios. To this end, we propose a novel model-based multi-view visual tracking algorithm. The algorithm integrates a predictive model of DLO behavior based on the Cosserat rod formulation and employs a neural network-based approximation to enable efficient evaluation of DLO shapes. By guiding visual perception with the predictive model, the algorithm effectively manages occlusions and estimates the 3D shape of the manipulated DLO in cluttered environments. This is accomplished by triangulating simple 2D images, enabling seamless integration into existing robotic systems without the need for costly and often unreliable 3D sensors. The proposed method is evaluated in a real-world scenario, demonstrating its effectiveness in reliably tracking thin DLOs in 3D environments.**

***Index Terms*—deformable linear objects, visual tracking, multi-view triangulation, robotic manipulation**

I. INTRODUCTION

Deformable Linear Objects (DLOs), such as electrical cables, wires, and hoses, are long, flexible objects with circular cross-sections [1]. Their manipulation is vital in industries like automotive, aerospace, and switchgear assembly, where tasks like routing wires are still largely manual, labor-intensive, and error-prone [2], [3]. Automating DLO handling is challenging due to their deformability, small size, and the need for precise perception, tracking, occlusion handling, and shape control during manipulation [2], [4], [5].

Recent research has introduced various methods to improve robotic perception and manipulation of DLOs. Learning-based models are favored for their real-time prediction capabilities and efficiency over traditional analytical models [6], [7]. Perception techniques include deep segmentation networks [8], stereo vision [9], and point cloud-based tracking [10]–[12], often enhanced with learning algorithms [13], [14]. However, these methods face limitations in real-world settings [3], such as difficulty handling dynamic scenes, reliance on pre-segmented data, poor occlusion handling, and lack of temporal continuity in shape estimation. Most are tested in controlled environments, limiting their practical applicability.

This work introduces a novel multi-view, model-based approach for visual tracking of DLOs during robotic manipulation (Fig. 1). By using simple 2D images from multiple camera angles, the method triangulates and fuses data to estimate the DLO's 3D shape, offering advantages over traditional 3D sensors [15]. It incorporates a fast neural network approximation of a Cosserat rod-based model to predict shape changes online, guiding perception and handling occlusions effectively in dynamic environments.

Alessio Caporali and Gianluca Palli are with DEI - Department of Electrical, Electronic and Information Engineering, University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy.

A. Caporali is funded by FSE+ 2021-2027 under a research contract per Law 240/2010, Art. 24(3)(a), and D.G.R. 693/2023 (REF. PA: 2023-20090/RER - CUP: J19J23000730002).

Corresponding author: alessio.caporali@unibo.it

II. METHOD

A. DLO Model

The DLO model employed in this work is based on the Cosserat rod theory, which describes the DLO as a thin, flexible, and extensible rod [16]. While the Cosserat rod model is accurate and realistic, it is computationally intensive, making it unsuitable for online robotic applications. To overcome this, a NN is trained for its approximation.

1) Analytical Cosserat Rod Model: A Cosserat rod is described by its centerline $s(z,t)$ (where z is the arc length of the rod and t is time) and a material frame. Details of the model and its governing equations are provided in [16]. The robot's manipulation actions are applied to the rod's extremities. Each extremity is associated with a pose vector, defined by the vertex position s and the material frame Q. The action is described as a displacement and a rotation applied to the current poses of the extremities. The action set is represented as $\mathcal{A} = \{a_1, a_n\}$, where a_1 and a_n refer to the action on the first and last DLO ends. Considering for reference the action $a_1 \in \mathbb{R}^7$, it is defined as $a_1 = [\delta_x, \delta_y, \delta_z, q_x, q_y, q_z, q_w]^\top$, where δ_x, δ_y and δ_z are the linear displacements applied to vertex s_1 and q_x, q_y, q_z and q_w are the quaternion components representing the rotation applied to the material frame Q_1. A similar definition holds for a_n.

2) Neural Network Model: A NN approximates the Cosserat rod model, offering a computationally efficient predictive model. To simplify the learning process, the DLO state is reduced and represented as a sequence of 3D points, each corresponding to a vertex s_i of the rod discretization $\mathcal{S} = \{s_1, \dots s_n\}$. The NN is trained to predict state changes caused by a given action $\mathcal{A}$.

The architecture of the NN (Fig. 2) is derived from [6] with several modifications to accommodate the 3D nature of the DLO state and the differences in action parameters. The network consists of a series of linear layers, each followed by a ReLU activation function. The network's output, denoted as $\tilde{\mathcal{S}}$, represents the predicted changes in the 3D coordinates of the DLO from the initial state. The final DLO state $\mathcal{S}_{\text{pred}}$ is then obtained by adding $\tilde{\mathcal{S}}$ to $\mathcal{S}_{\text{in}}$, i.e.:

$$\mathcal{S}_{\text{pred}} = \texttt{DloPredictiveModel}(\mathcal{S}_{\text{in}}, \mathcal{A}) = \tilde{\mathcal{S}} + \mathcal{S}_{\text{in}}.$$

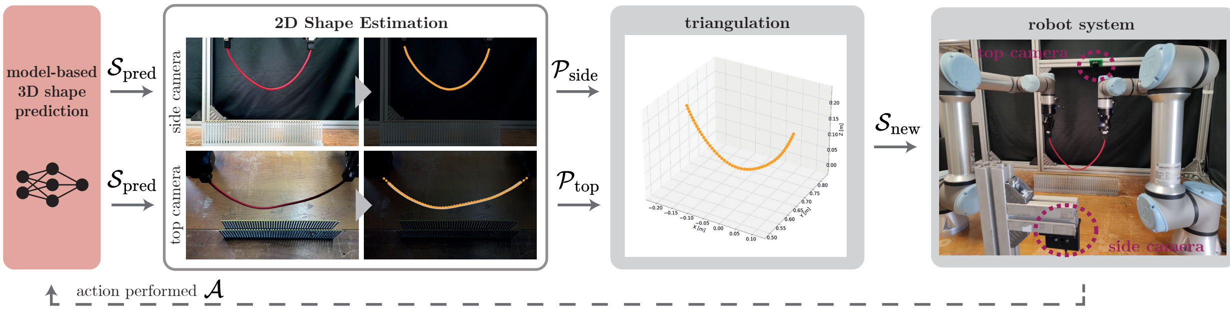

Fig. 1: Overview of the proposed multi-view model-based tracking approach.

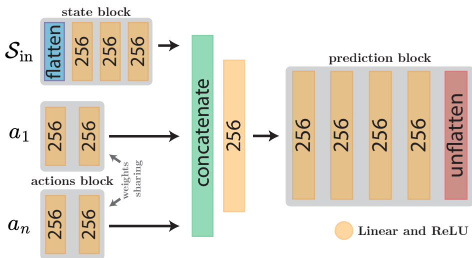

Fig. 2: Neural network architecture.

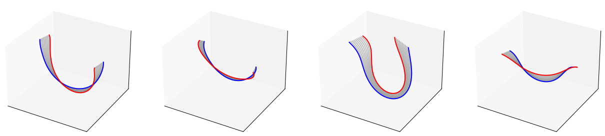

Fig. 3: Dataset samples generated by the Cosserat rod model.

The network is trained to minimize the mean squared error between the predicted $\mathcal{S}_{\text{pred}}$ and the expected $\mathcal{S}_{\text{out}}$ final states.

The dataset is generated by simulating the analytical DLO model subjected to a series of random actions, see Fig. 3.

B. Multi-View Visual Perception

The vision system employed in this work is based on two 2D cameras, a *side-view* and a *top-view* camera, which provide images of the manipulated DLO from different perspectives.

1) 2D Shape Estimation: The 2D shape estimation is performed independently for each camera. The process begins with detecting the target manipulated DLO in the image, utilizing a graph-based approach inspired from *RT-DLO* [17]. First, a semantic segmentation network is first employed to remove the background from the image [8], Next, a specialized pipeline is applied to extract the target DLO by leveraging an initial guess based on the predicted DLO state $\mathcal{S}_{\text{pred}}$.

The pipeline consists of three key steps: 1) projecting $\mathcal{S}_{\text{pred}}$ onto each camera's view obtaining $\mathcal{P}_{\text{pred}}$, 2) generating a graph representation of the DLO as observed in the scene [17], and 3) associating the nodes of the graph with $\mathcal{P}_{\text{pred}}$.

The association process is illustrated in Fig. 4. It is performed by matching the projected points p_i with the nodes v_j of the graph. Importantly, the association is guided by the predicted DLO state, such that erroneous or missing nodes do not affect the matching process. Considering a point p_i, the process begins by identifying its closest edge (v_1, v_2) in the graph. The point p_i is then projected onto this edge, yielding

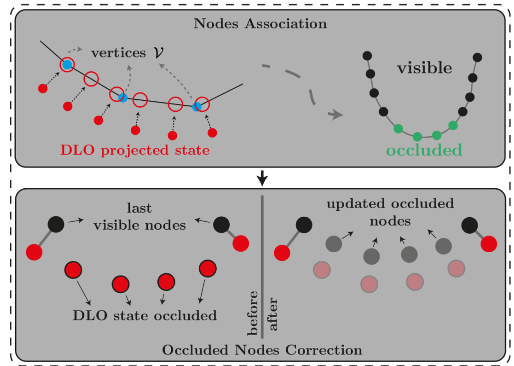

Fig. 4: Node association procedure.

a projected point p'_i. The projection is computed as follows:

$$p'_i = v_1 + \left(\frac{(p_i - v_1) \cdot (v_2 - v_1)}{\|v_2 - v_1\|^2} \right) (v_2 - v_1)$$

This *edge projection* ensures a more accurate association compared to using the closest vertex only. Indeed, the DLO is expected to lie along the edges of the graph, and the number and distribution of the vertices sampled with FPS may differ significantly from $\mathcal{P}_{\text{pred}}$.

By repeating this process for all points p_i, the association between $\mathcal{P}_{\text{pred}}$ and $\mathcal{V}$ is established. Simultaneously, the segmentation mask M_b is used to assign a *visible* or *occluded* attribute to each point p_i in $\mathcal{P}_{\text{pred}}$ by evaluating the corresponding pixel value of p'_i in M_b.

For the *occluded* points, a reliable association is therefore not possible, and the original predicted DLO state $\mathcal{S}_{\text{pred}}$ is used as a fallback. However, a gap (or step) is usually introduced between the $\mathcal{P}_{\text{pred}}$ and the associated points, as the predicted DLO state is not perfectly aligned with the graph. To provide as output a smooth DLO state, the gap is addressed by translating the occluded points over the expected centerline of the occluded area. This translation is performed by interpolating between the *edge projection* distances, i.e. the distance between p_i and p'_i, of the associated points at the extremity of the occluded areas. This approach ensures that the vision system can handle occlusions effectively, as the NN model provides an estimate of the DLO's state in these cases.

Thus, the final output is a set of 2D points $\mathcal{P}_{\text{final}}$ representing the DLO's shape in the image.

2) Multi-view Triangulation: Triangulation is the process of determining the 3D position of a point by intersecting the rays passing through it from different points of view. In the

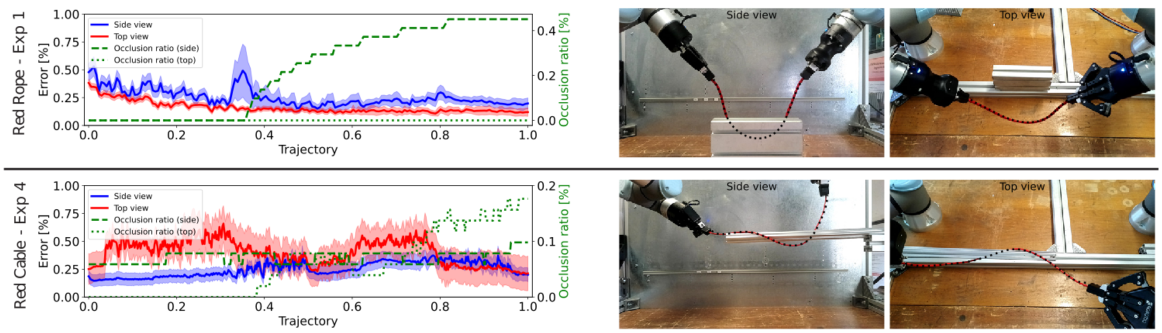

Fig. 5: Example of tracking performance in real-world scenarios for different experiment trajectories and DLOs.

TABLE I: Average DTC error for real-world tracking across DLOs and trajectories (values in percentage).

Exp Setup	**Red Rope**	**Red Cable**	**Blue Cable**
Real Exp 1	0.217	0.217	0.234
Real Exp 2	0.200	0.187	0.203
Real Exp 3	0.213	0.172	0.231
Real Exp 4	0.317	0.324	0.379

context of the DLO, the 2D shapes obtained from the *side-view* and *top-view* cameras are triangulated to reconstruct the 3D shape of the DLO.

Given a pixel point p in the 2D image plane of the camera, the unit ray ν passing through the image reference frame origin and p can be expressed in the camera frame as:

$$\nu' = \begin{bmatrix} p_x - c_x \\ p_y - c_y \\ f \end{bmatrix}, \quad \nu = \frac{\nu'}{\|\nu'\|}$$

where c_x and c_y are the pixel coordinates of the image center and f is the camera focal distance.

The unit ray ν can be expressed in a reference world frame by ${}^w\nu = {}^wT_c v$ where ${}^wT_c$ is the transformation matrix from the camera frame to the world frame.

Provided that m distinguished points of view are available, the estimation $\tilde{p}$ of the unknown point p can be obtained by looking for the point having the minimum distance from all the rays. By defining the symmetric matrix $V_i = I - {}^w\nu_i {}^w\nu_i^T$, providing the semi-norm on the ray distance, the point location estimate $\tilde{p}$ is provided by the nearest point search algorithm:

$$\tilde{p} = \left(\sum_{i=1}^{m} V_i \right)^{-1} \left(\sum_{i=1}^{m} V_i \, {}^w t_{c_i} \right)$$

where ${}^w t_{c_i}$ is the translation vector of the camera i in the world frame, $i = 1, \ldots, m$.

This procedure is thus applied to each DLO node.

III. Experiments

The experimental setup includes two UR5e robots with two-fingered grippers (Robotiq Hand-E, 2F-85) and two static 2D cameras (Luxonis OAK-1, 1920×1080 pixels) providing *side* and *top* views. Both cameras are intrinsically and extrinsically calibrated to the robot bases. ROS2 handles communication, and the algorithm, implemented in Python 3.10 with PyTorch 2.0, runs on a workstation with an Intel Core i9-9900K CPU (3.60 GHz) and an NVIDIA GTX 2080 Ti GPU.

The tracking algorithm's performance is evaluated using: 1) Occlusion Ratio, defined as the percentage of DLO state points that are obscured in image space; 2) Distance-to-Centerline (DTC), defined in the image space as the shortest distance between the predicted projected state and the GT centerline in image space (the error is normalized by the image width and expressed as a percentage).

Three different real-world DLOs are used: red rope (length 0.53 m, diameter 6.0 mm); blue cable (length 0.60 m, diameter 4.8 mm); and red cable (length 0.52 m, diameter 3.6 mm). The selection of these DLOs is based on their varying flexible behaviors. The red rope and red cable share similar flexibility characteristics but differ significantly in diameter and material. The blue cable is included to test the tracking algorithm with a stiffer DLO that exhibits pronounced plastic deformation. Four distinct manipulation trajectories are used with the DLOs undergoing varying motions and levels of occlusion.

A. Real-World Tracking Algorithm Results

Ground truth centerlines are obtained by manually annotating DLO masks in each frame and extracting the visible centerline through skeletonization. Tracking accuracy is then quantitatively evaluated using the DTC metric.

Quantitative results of the real-world experiments are provided in Tab. I. The DTC error is computed for each experiment, demonstrating the algorithm's ability to track the DLO state in real-world scenarios accurately. The tracking performance remains consistent across the diverse set of test DLOs and trajectories, with the average DTC error staying below 1% in all cases. This is illustrated in the plots of Fig. 5, which display the error progression during trajectory execution for various combinations of DLO and manipulation trajectory. The figure also includes snapshots of the various trajectories and occlusion setups involved. Notably, the same NN predictive model is employed across all test DLOs without any DLO-specific fine-tuning.

IV. Conclusions

This paper introduces a new visual tracking method for DLOs, combining a fast neural network-based predictive model with a multi-view triangulation approach. The method effectively tracks DLOs even under occlusions. Tested in real-world experiments with various DLO types, the system runs at 15 Hz, supporting real-time feedback. Future work aims to improve the predictive model and extend the approach to more complex manipulation and collision scenarios.

REFERENCES

[1] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The Int. J. of Robotics Research*, 2018.

[2] J. Trommnau, J. Kühnle, J. Siegert, R. Inderka, and T. Bauernhansl, "Overview of the state of the art in the production process of automotive wire harnesses, current research and future trends," *Procedia CIRP*, 2019.

[3] G. Palli, S. Pirozzi, M. Indovini, D. De Gregorio, R. Zanella, and C. Melchiorri, "Automatized switchgear wiring: An outline of the wires experiment results," *Advances in Robotics Research: From Lab to Market: ECHORD++: Robotic Science Supporting Innovation*, 2020.

[4] Y. Gao, Z. Chen, Y. Ling, J. Yang, Y.-H. Liu, and X. Li, "A hierarchical manipulation scheme for robotic sorting of multiwire cables with hybrid vision," *IEEE/ASME Transactions on Mechatronics*, 2022.

[5] W. Ma, B. Zhang, L. Han, S. Huo, H. Wang, and D. Navarro-Alarcon, "Action planning for packing long linear elastic objects into compact boxes with bimanual robotic manipulation," *IEEE/ASME Transactions on Mechatronics*, 2022.

[6] 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*, 2024.

[7] 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*, 2022.

[8] A. Caporali, M. Pantano, L. Janisch, D. Regulin, G. Palli, and D. Lee, "A weakly supervised semi-automatic image labeling approach for deformable linear objects," *IEEE Robotics and Automation Letters*, 2023.

[9] A. Caporali, K. Galassi, and G. Palli, "Deformable linear objects 3D shape estimation and tracking from multiple 2D views," *IEEE Robotics and Automation Letters*, 2023.

[10] Y. Wang, D. McConachie, and D. Berenson, "Tracking partially-occluded deformable objects while enforcing geometric constraints," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[11] T. Tang and M. Tomizuka, "Track deformable objects from point clouds with structure preserved registration," *The International Journal of Robotics Research*, 2022.

[12] J. Xiang, H. Dinkel, H. Zhao, N. Gao, B. Coltin, T. Smith, and T. Bretl, "Trackdlo: Tracking deformable linear objects under occlusion with motion coherence," *IEEE Robotics and Automation Letters*, 2023.

[13] K. Lv, M. Yu, Y. Pu, X. Jiang, G. Huang, and X. Li, "Learning to estimate 3-d states of deformable linear objects from single-frame occluded point clouds," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023.

[14] Y. Yang, J. A. Stork, and T. Stoyanov, "Particle filters in latent space for robust deformable linear object tracking," *IEEE Robotics and Automation Letters*, 2022.

[15] K. P. Cop, A. Peters, B. L. Žagar, D. Hettegger, and A. C. Knoll, "New metrics for industrial depth sensors evaluation for precise robotic applications," in *IEEE Int. Conf. IROS*, 2021.

[16] M. Gazzola, L. Dudte, A. McCormick, and L. Mahadevan, "Forward and inverse problems in the mechanics of soft filaments," *Royal Society open science*, 2018. [Online]. Available: https://doi.org/10.1098/rsos.171628

[17] 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 Trans. on Industrial Informatics*, 2023.