Learn to Predict Posterior Probability in Particle Filtering for Tracking Deformable Linear Objects

Yuxuan Yang[1], Johannes A. Stork[1], and Todor Stoyanov[1,2]

***Abstract*— Tracking deformable linear objects (DLOs) is a key element for applications where robots manipulate DLOs. However, the lack of distinctive features or appearance on the DLO and the object's high-dimensional state space make tracking challenging and still an open question in robotics. In this paper, we propose a method for tracking the state of a DLO by applying a particle filter approach, where the posterior probability of each sample is estimated by a learned predictor. Our method can achieve accurate tracking even with no prerequisite segmentation which many related works require. Due to the differentiability of the posterior probability predictor, our method can leverage the gradients of posterior probabilities with respect to the latent states to improve the motion model in the particle filter. The preliminary experiments suggest that the proposed method can provide robust tracking results and the estimated DLO state converges quickly to the true state if the initial state is unknown.**

I. Introduction

Robotic manipulation of deformable linear objects (DLOs) has potential for a wide range of real-world applications [1], [2], such as surgical suturing [3] in medical robotics and cables and hoses handling [4], [5] in industrial robotics. However, these tasks are still challenging due to the difficulties in dynamics modeling and state tracking of DLOs [1]. This paper focuses on the latter challenge.

Deformable linear objects have a high-dimension state space. They are conventionally described as a chain of nodes and the state is represented by the set of nodes' positions. This is one of the major factors that differentiate tracking DLOs from rigid objects where many off-the-shelf tracking methods are available. Particle filtering is a common solution for rigid object tracking, while it cannot be directly applied to DLO state tracking because of the large number of particles it needs for sampling in the high-dimensional state space. For instance, Lagneau et al. [6] represent a DLO by a few control points of B-splines so that a particle filter is feasible for only tracking the position of these control points. Our previous work [7] uses a particle filter to track a DLO by sampling in a learned low-dimensional latent state space.

Regardless of the success in previous works about DLOs state tracking [2], the assumptions and limitations in current tracking methods hinder the application of those methods in real-world tasks. One major assumption is that color-based segmentation is available for generating masks for DLOs on RGB images [8]–[10]. Yan et al. [11] alleviate the assumption of a known DLO and background colors but good color contrast between the DLO and the background is still necessary. Learning-based instance segmentation has recently become an option for the semantic segmentation of DLOs since a DLO dataset generation approach is presented and a dataset is publicly available [12], otherwise annotating a large set of images is a challenging task. Based on this dataset, Ariadne [13] and an improved version Ariadne+ [14] are developed for instance segmentation of DLOs. However, they might suffer from a sim-to-real gap and they are not able to provide a robust and complete solution in some cases [14].

In this paper, we propose a method that uses a particle filter to track a DLO by sampling in the latent state space and a data-driven posterior probability predictor. The overview of the posterior probability prediction process is shown in Fig. 1. We leverage learning-based instance segmentation and train a neural network to predict the posterior probability of the latent state samples given an observation instead of

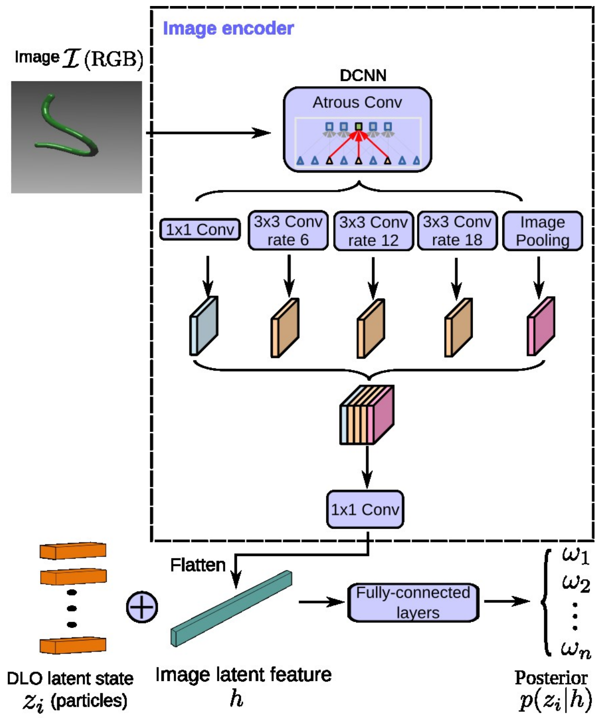


Fig. 1: Overview of the posterior probability prediction in our particle filtering tracking framework.

[1] Autonomous Mobile Manipulation Lab, Center for Applied Autonomous Sensor Systems (AASS), Örebro University, Sweden.

[2] Department of Computing and Software, McMaster University, Canada.

The corresponding author: yuxuan.yang@oru.se.

This work was partially supported by Vinnova / SIP-STRIM projects 2019-05175, and was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

color-based segmentation and handcraft posterior probability calculation implemented in our previous work [7]. Therefore, we make the following contributions: first, the method does not rely on color-based segmentation; second, more particles are allowed because the neural-network-based predictor can compute all posterior probabilities in parallel, leading to the potential for multiple DLOs tracking; third, by introducing the gradients of posterior probabilities with respect to the sampled latent states from the predictor, we improve the motion model which accelerates the convergence of DLO state estimation if the initial DLO state is unknown.

II. Background

In this paper, a deformable linear object (DLO) is discretized into N nodes, $\mathcal{X}^t$, where $\mathcal{X}^t = \{\boldsymbol{x}_1^t, \boldsymbol{x}_2^t, ..., \boldsymbol{x}_N^t\}$, and describe the DLO state at time t as a sequence of these nodes' positions, $\boldsymbol{X}^t = [{\boldsymbol{x}_1^t}^\intercal, {\boldsymbol{x}_2^t}^\intercal, ..., {\boldsymbol{x}_N^t}^\intercal]^\intercal$, where $\boldsymbol{x}_i^t \in \mathbb{R}^3$ is the position of the ith node at the time step t. The DLO state is the output of the tracking algorithm given a sequence of RGBD images $\mathcal{I}^{1:t}$. We assume a point cloud is generated from the RGBD image. Our goal in tracking is to minimize $\|\hat{\boldsymbol{X}}^t - \boldsymbol{X}^t\|_2$, where $\hat{\boldsymbol{X}}^t$ is the tracking result and $\boldsymbol{X}^t$ is the ground truth state.

III. Method

In this section, we describe our DLO state tracking method based on Sequential Importance Resampling (SIR) Particle Filter with a posterior probability predictor. First, we explain low-dimensional latent DLO state representation learning (III-A). Then, we introduce the encoder for image latent feature extracting (III-B). Last, we formulate the tracking problem in a particle filtering framework and describe in detail the motion model and the posterior probability predictor (III-C).

A. DLO's Latent Feature

We take a data-driven approach to construct a low-dimensional embedding space, along with two mappings to translate to and from the learned manifold. Positions of the nodes along a DLO is highly correlated and our goal is to find a low-dimensional latent state space that can represent the DLO state, $\boldsymbol{X}^t = [{\boldsymbol{x}_1^t}^\intercal, {\boldsymbol{x}_2^t}^\intercal, ..., {\boldsymbol{x}_N^t}^\intercal]^\intercal$, which forms a manifold within $\mathbb{R}^{3N}$. Then it becomes practical to use a particle filter in the low-dimensional state space instead of the high-dimensional one. To this end, we employ an autoencoder that consists of two parametric functions—an encoder, $\phi : \mathbb{R}^{3\times N} \rightarrow \mathbb{R}^m$, and a decoder, $\psi : \mathbb{R}^m \rightarrow \mathbb{R}^{3\times N}$, where m is the dimension of the latent state. The autoencoder is trained through unsupervised learning. With trained parameters, for a given state, $\boldsymbol{X}$, we have

$$\boldsymbol{z} = \phi(\boldsymbol{X}) \tag{1}$$

$$\boldsymbol{X}' = \psi(\boldsymbol{z}) \tag{2}$$

where $\boldsymbol{z}$ is the latent state for the particle-filter-based tracking and $\boldsymbol{X}'$ is the reconstructed state. The goal is to reconstruct a $\boldsymbol{X}'$ that is as close as possible to the original $\boldsymbol{X}$.

In this paper, we use a multi-layer perceptron to implement the encoder and the decoder with a symmetric structure. The dimension m of the latent state $\boldsymbol{z}$ is a customizable parameter where a higher dimension usually leads to a lower reconstruction error but increases the computational burden for the particle-filter-based tracking at the same time.

B. Image's Latent Feature

To extract the latent feature of the image, we employ the encoder of DeepLabv3+ neural network model [15]. This image encoder is shown in Fig. 1.

$$\boldsymbol{h} = g(\mathcal{I}) \tag{3}$$

The DeeplabV3+ represents the state-of-the-art in semantic segmentation and is capable to provide a binary mask of the wires (DLOs) in Ariadne+ [12], [14]. Its encoder module can encode multi-scale contextual information by applying atrous spatial pyramid pooling at multiple scales. We use the trained model parameters from Ariadne+ [14] which is trained on the Electric Wires Image Segmentation dataset [12], a dataset of approximate 30,000 synthetic images of wires with various shapes and colors in different backgrounds.

C. Particle Filtering

A particle filter uses a collection of particles to represent the posterior distribution of a stochastic process given noisy and/or partial observations [16]. At each time step, particles evolve according to a motion model, or transition probability density $p(\boldsymbol{z}_i^t|\boldsymbol{z}_i^{t-1})$. Each particle is assigned a weight, ω_i, according to the posterior probabilities—i.e., the probabilities of each proposed latent state given the current observation. Then the particles are resampled based on the importance weights, where particles with higher weights are more likely to be preserved to the next time step. The filtering probability density is estimated by a weighted sum of all particles.

Implementing a particle filter directly for DLO state tracking requires us to estimate a posterior distribution of a DLO state, given the current observation — $p(\hat{\boldsymbol{X}}^t|\mathcal{I}^t)$. However, the particle filter cannot handle this high-dimensional state space. In our previous work [7], we implemented a particle filter based on the posterior distribution of latent DLO state given the observation, $p(\boldsymbol{z}^{\boldsymbol{t}}|\mathcal{I}^t)$. In this paper, instead of relying on color-based segmentation and a complex registration-based approach to calculate posterior probabilities, we learn a posterior probability predictor through data, which predicts posterior probabilities of latent states given the image feature from an observation, $p(\boldsymbol{z}^{\boldsymbol{t}}|\boldsymbol{h})$. The overview of the predictor is shown in Fig. 1, and the tracking process is described in Algorithm 1.

In this paper, we relax the assumption of knowing the initial DLO state compared to some related works [8], [10], [17]. If the initial state is unknown, the particle filtering starts with randomly sample particles in the latent space. Because the predictor is differentiable, gradients with respect to the DLO latent space can be used to quickly push particles to the local maximum, accelerating the convergence to a good estimation of the posterior distribution. If the initial

Algorithm 1: Tracking DLO state by a particle filter

Input : $\mathcal{I}^t$,$\boldsymbol{z}_{1:K}^{t-1}$, α, $\boldsymbol{\Sigma}$

Output: $\boldsymbol{z}_{1:K}^t$, $\omega_{1:K}^t$

$\hat{\boldsymbol{z}}_{1:K}^t \leftarrow \mathrm{N}(\boldsymbol{z}_{1:K}^{t-1} + \alpha(\boldsymbol{z}_{1:K}^{t-1} - \boldsymbol{z}_{1:K}^{t-2}), \boldsymbol{\Sigma})$

$\boldsymbol{h} = g(\mathcal{I}^t)$ # Image encoder (Equation (3))

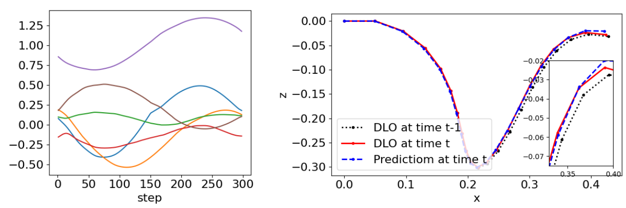
for $i \leftarrow 1$ **to** K **do**

  $\omega_i^t = f_p(\boldsymbol{z}_i^t, \boldsymbol{h})$;

end

$\omega_{1:K} = \frac{\hat{\omega}_{1:K}}{\sum \hat{\omega}_{1:K}}$

$\boldsymbol{z}_{1:K}^t \leftarrow$ `resample`$(\boldsymbol{z}_{1:K}^t, \omega_{1:K}^t)$ # Equation 4 and 5



(a) (b)

Fig. 2: (a): Changes of 6-dimension latent state when a DLO moves; (b): The DLO state prediction (projected in x-z plane) via the motion model in the latent state.

state is known, we pass it through the encoder and use its latent state to initialize all particles. After initialization, we first propagate the K particles according to the motion model. Then, we predict the posterior probability for each particle, $p(\boldsymbol{z}_i^t|\mathcal{I}_t)$, and update the weights, ω_i. In the end, we resample particles by systematic resampling to avoid the degeneracy problem of the algorithm. This resampling method is favorable due to its computational complexity and good empirical performance [18]. Systematic resampling generates K ordered numbers u_k and uses them to select samples $\boldsymbol{z}_{1:K}^t$ by the multinomial distribution.

$$u_k = \frac{(k-1)+\tilde{u}}{K}, \text{with } \ \tilde{u} \sim \mathrm{U}[0,1) \tag{4}$$

$$\boldsymbol{z}_k^t = \boldsymbol{z}_i^{t-1} \text{ with } i \text{ s.t.} u_k \in [\sum_{s=1}^{i-1} \omega_s, \sum_{s=1}^{i} \omega_s] \tag{5}$$

Next, we explain the two main components in our particle filter—the motion model and the posterior probability.

A motion model propagates the distribution of the state from the previous time step $t-1$ to the current time step t. Empirically we note that under nominal motion of the DLO, the latent encoding changes smoothly, as shown in Fig. 2a. Therefore, we use a constant velocity model to approximate the probability distribution propagation of the latent state.

$$p(\boldsymbol{z}_i^t|\boldsymbol{z}_i^{t-1}, \boldsymbol{z}_i^{t-2}) = \mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{6}$$

where $\mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate normal distribution with mean $\boldsymbol{\mu} = \boldsymbol{z}_i^{t-1} + \alpha(\boldsymbol{z}_i^{t-1} - \boldsymbol{z}_i^{t-2})$ and covariance matrix $\boldsymbol{\Sigma}$. Both $\boldsymbol{\Sigma}$ and α are hyperparameters. The motion model helps approximate the transition in the latent state and an example is shown in Fig. 2b. We assume $\boldsymbol{\Sigma} = \beta^2 \boldsymbol{I}_n$, where $\boldsymbol{I}_n$ is an n-dimension identity matrix. When $\alpha = 0$, the distribution propagation is only driven by injecting isotropic Gaussian noises with variance β^2. The motion model can be improved by the gradient information from the posterior probability predictor, which will be introduced later.

Posterior probability, $p(\boldsymbol{z}_i^t|\boldsymbol{h}^t)$, measures how likely a latent state, $\boldsymbol{z}_i^t$, is given the image feature, $\boldsymbol{h}^t = g(\mathcal{I}_t)$, from an observation. Instead of using a registration-based approach to calculate the posterior probabilities in our previous work [7], we directly use a learned predictor to predict $p(\boldsymbol{z}_i^t|\boldsymbol{h}^t)$ given the observation and proposed latent states, $\omega_i = f_p(\boldsymbol{z}_i, \boldsymbol{h}^t)$.

The posterior probability predictor, f_p, is fully-connected layers. We concatenate the image feature and the sampled latent state as the input of the predictor. The predictor is trained using the supervised learning approach, where the ground truth posterior probability is the reciprocal of the difference between the proposed DLO state, $\hat{\boldsymbol{X}}_i^t$, and the ground truth DLO state, $\boldsymbol{X}^t$, as the following:

$$d_i = \|\hat{\boldsymbol{X}}_i^t - \boldsymbol{X}^t\|_2 \tag{7}$$

$$\omega_i = \frac{\hat{\omega}_i}{\sum \hat{\omega}_i}, \quad \hat{\omega}_i = \frac{1}{d_i} \tag{8}$$

where $\hat{\boldsymbol{X}}_i^t$ is decoded from the sampled latent state, $\hat{\boldsymbol{X}}_i^t = \psi(\boldsymbol{z}_t)$.

Revised motion model The posterior probability predictor, f_p, is differentiable, therefore, the gradients of a posterior probability with respect to the latent states, $\frac{\partial f_p}{\partial \boldsymbol{z}_i}$, are available. We can use this gradient information to improve the simple motion model shown in Equation (6). The mean in the multivariate normal distribution can be modified as

$$\boldsymbol{\mu} = \boldsymbol{z}_i^{t-1} + \alpha(\boldsymbol{z}_i^{t-1} - \boldsymbol{z}_i^{t-2}) + \beta \frac{\partial f_p}{\partial \boldsymbol{z}_i^{t-1}} \tag{9}$$

where β is the step size on the gradient direction.

IV. Preliminary Results

Preliminary experiments in simulation show the qualitative results of DLO state tracking and the improvement of the convergence of posterior estimation by employing the gradients of the posteriors with respect to the proposed latent states.

We implement the posterior probability predictor as a 5-layer fully-connected network with $[1024, 512, 256, 128, 64]$ units in each layer. We choose LeakyReLU as the activation function with a 0.01 negative slope. We implement the model f_p in PyTorch [19]. Adam [20] is the optimizer with a fixed learning rate 1×10^{-3}. In the preliminary experiments, we generate synthetic data to train the predictor. We use 500 particles in particle filtering.

Qualitative results of tracking performance using particle filtering based on the learned posterior predictor are shown in Fig. 3. This is a proof of concept showing our proposed tracking method works in a simplified scenario. If the initial state of the DLO is unknown, the particle filter can gradually converge to an accurate estimation of the DLO state after several time steps. With the gradient information of the posteriors with respect to the latent state, the convergence is faster. A qualitative comparison is shown in Fig. 4.

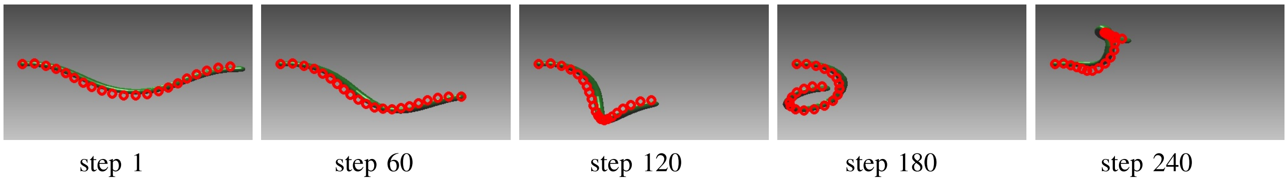


Fig. 3: Qualitative results of tracking performance.

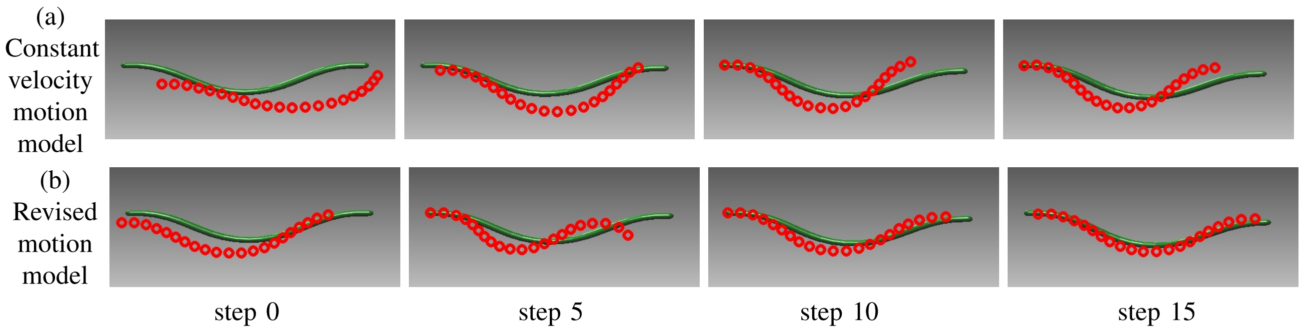


Fig. 4: Qualitative comparison of the DLO state estimation convergency when the initial DLO state is unknown. The estimated state converge faster by using revised motion model where the gradients of posterior with respect to the sampled DLO states are employed.

V. Discussion

We propose a method that tracks a deformable linear object (DLO) using a particle filter based on a posterior probability predictor. The predictor we introduced to the particle filtering framework from our previous work [7] offers three advantages: first, it releases the color-based segmentation requirement, which is an assumption and limitation for most related works [7], [8], [17]; second, with this neural-network-based predictor, it is straightforward to compute all posterior probabilities in parallel which leads to the potential of introducing more particles for multiple DLOs state tracking; third, by leveraging the differentiability of the posterior probability predictor, we can improve the motion model by using the gradients with respect to the sampled latent state to accelerate the convergency of the DLO state estimation if the initial state is unknown.

Our preliminary experiments suggest that the proposed method provides robust tracking results in a simplified scenario and the gradient information from the posterior probability predictor improves the motion model. We will further improve and test the method for multiple DLOs tracking and then apply this tracking method to multiple DLOs manipulation tasks such as using a robot to braiding two DLOs.

References

[1] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcon, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, X. Li, J. Pan, *et al.*, "Challenges and outlook in robotic manipulation of deformable objects," *arXiv preprint arXiv:2105.01767*, 2021.

[2] 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," *International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[3] M. Saha and P. Isto, "Manipulation planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1141–1150, 2007.

[4] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, and A. Cherubini, "Dual-arm robotic manipulation of flexible cables," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2018, pp. 479–484.

[5] Y. Yang, J. A. Stork, and T. Stoyanov, "Learning to propagate interaction effects for modeling deformable linear objects dynamics," in *Proc. of the IEEE International Conference on Robotics and Automation*. IEEE, 2021, pp. 1950–1957.

[6] R. Lagneau, A. Krupa, and M. Marchal, "Automatic shape control of deformable wires based on model-free visual servoing," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5252–5259, 2020.

[7] Y. Yang, J. A. Stork, and T. Stoyanov, "Particle filters in latent space for robust deformable linear object tracking," *under review*, 2022.

[8] Y. Wang, D. McConachie, and D. Berenson, "Tracking partially-occluded deformable objects while enforcing geometric constraints," in *Proc. of the IEEE International Conference on Robotics and Automation*. IEEE, 2021, pp. 14 199–14 205.

[9] S. Jin, C. Wang, and M. Tomizuka, "Robust deformation model approximation for robotic cable manipulation," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2019, pp. 6586–6593.

[10] S. Jin, C. Wang, X. Zhu, T. Tang, and M. Tomizuka, "Real-time state estimation of deformable objects with dynamical simulation," in *Workshop on Robotic Manipulation of Deformable Objects*, 2020.

[11] 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.

[12] 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.

[13] D. D. Gregorio, G. Palli, and L. D. Stefano, "Let's take a walk on superpixels graphs: Deformable linear objects segmentation and model estimation," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 662–677.

[14] A. Caporali, R. Zanella, D. De Gregorio, and G. Palli, "Ariadne+: Deep learning-based augmented framework for the instance segmentation of wires," *IEEE Transactions on Industrial Informatics*, 2022.

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[16] P. Del Moral, "Nonlinear filtering: Interacting particle resolution," *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, vol. 325, no. 6, pp. 653–658, 1997.

[17] C. Chi and D. Berenson, "Occlusion-robust deformable object tracking without physics simulation," in *Proc. of the IEEE/RSJ International*

Conference on Intelligent Robots and Systems. IEEE, 2019, pp. 6443–6450.

[18] R. Douc and O. Cappé, "Comparison of resampling schemes for particle filtering," in *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.* IEEE, 2005, pp. 64–69.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, pp. 8026–8037, 2019.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980