Wire Point Cloud Instance Segmentation from RGBD Imagery with Mask R-CNN

Holly Dinkel*[§], Jingyi Xiang[†][§], Harry Zhao*, Brian Coltin[‡], Trey Smith[‡], and Timothy Bretl*
*Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL
[†]Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL
[‡]Intelligent Robotics Group, NASA Ames Research Center, Moffett Field, CA
*[†]{hdinkel2,jingyix4,harryz2,tbretl}@illinois.edu, [‡]{brian.coltin,trey.smith}@nasa.gov

Abstract— **Perception of the shapes of deforming objects like wires enables their monitoring and manipulation by autonomous robots. This paper presents detection, classification, and instance segmentation of deformable wires from a cluttered scene in RGBD imagery. This work uses the Detectron2 implementation of Mask R-CNN trained with the PointRend mask head on the UIUCWires dataset as the framework for wire instance segmentation on RGB imagery, a method demonstrated to perform well for the instance segmentation task in previous work. In this work, the instance bitmask is directly used to segment individual object point clouds, an important step toward wire shape representation for manipulation tasks.**

I. Introduction

The problem of segmenting deformable objects like wires in cluttered scenes is an important task in robotics with a wide array of applications like manufacturing and domestic services. Unlike rigid objects, any forces applied to deformable objects result in their translation in space as well as changes to their shapes [1]. Additionally, deformable objects have high-dimensional state spaces and complex nonlinear dynamics which make model-based estimates of their shapes challenging [2]–[5].

A growing body of work involves wire tracking and manipulation [6]–[10]. There is relatively little work focused on wire perception despite the importance of perception in both model-based and model-free wire tracking and manipulation research. Perception of wires for tracking and manipulation involves obtaining quality object segmentation masks as a pre-processing step. Segmentation masks are produced via two conventional methods: color, or intensity, thresholding and segmentation using neural network-based frameworks. Intensity thresholding exploits color contrasts between objects in the foreground and the background of an image. It conveniently requires neither data management nor neural network training for online deployment, however it is not robust to noise in background color or texture and to separating distinct objects in the foreground. These challenges considerably limit the scope of relevant applications; it is impractical for obtaining object instance masks in wire tracking and manipulation tasks in most real environments.

We are interested in the **3D wire object instance segmentation problem**. The ability to segment individual wire object instances, as opposed to generating one single semantic segmentation mask for all wires in the scene, is critical for enabling typical robotic wire inspection and manipulation tasks in the (usual) case that many wires appear together in the same scene. This work uses a two-step approach. First, the Detectron2 implementation of Mask R-CNN trained with the PointRend mask head on the UIUCWires dataset with object segment semantics produces binary segmentation masks of wires and ethernet devices in RGB images [11]–[13]. Next, the predicted binary segmentation masks are used to segment wires and devices in the corresponding depth image to extract individual object point clouds. This work uses both the Object Semantics (OS) and Object Segment Semantics (OSS) models from UIUCWires to perform depth segmentation. Under the OSS representation, Mask R-CNN is not required to associate various segments of a wire object across wire crossover points. This relaxes the problem from predicting full object masks to predicting unoccluded parts of these masks. In future work, these segment semantics could be combined with a post-processing step for explicit segment association.

II. Object Instance Segmentation

Point-wise semantic labeling of pointcloud data remains an open challenge in computer vision. In general, point data in depth imagery are more difficult to label to obtain ground truth information than pixel data in RGB imagery. This work leverages automatic RGB image labeling in the UIUCWires dataset to train Mask R-CNN to perform object instance segmentation in RGB imagery. The UIUCWires dataset is an RGB image dataset comprising 2,000 training images of up to four wires and up to two devices per image. Images in the dataset include two schemes of object segmentation mask representation. With *Object Semantics* (OS) mask representations, all contours comprising an object receive one corresponding bitmask. With *Object Segment Semantics* (OSS) mask representations, each object contour receives a unique bitmask and thus occlusions by other wires and devices is not explicitly represented [13]. The UIUCWires training data are primarily synthetically generated from single-object (single-wire and single-device) images using color thresholding techniques to automatically obtain object instance masks. It uses the copy-paste, background substitution, scale jittering, and hue jittering augmentation techniques to increase the va-

[§]Equal contribution

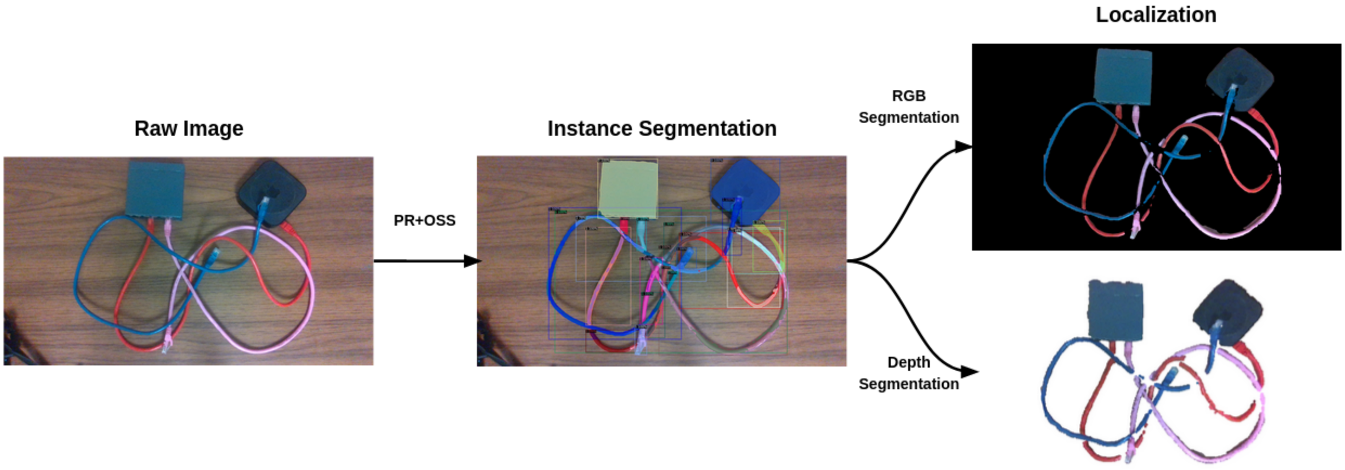

Fig. 1. In the proposed perception pipeline, an RGB image is segmented using Mask R-CNN with the PointRend (PR) mask head trained on the UIUCWires dataset with Object Segment Semantics (OSS) annotations. The predicted instance segmentation computed by Mask R-CNN is used to segment the RGB image ([13]) and its associated depth image to obtain a point cloud segmented by object instance (this work).

riety of wire configurations and environmental noise classes present in the dataset [14]. This method enables automatic labeling of images comprising wires of the same color or of the same color as the background.

Mask R-CNN is the state-of-the-art method of object instance segmentation [12]. It performs object detection (classification), localization (bounding-box prediction), and instance segmentation (instance-level mask prediction). It includes a two-stage training procedure. In the convolutional backbone, a region proposal network performs feature extraction and generates candidate bounding boxes [15]. The second stage includes the network *head* which performs classification and bounding box recognition in parallel with segmentation mask prediction over each region of interest. The Mask R-CNN architecture is optimized according to a multi-task loss on sampled regions of interest according to

$$L = L_{cls} + L_{box} + L_{mask}. \tag{1}$$

where L_{cls} is the log loss of the predicted class probability, L_{box} is the Smooth L1-loss optimizing the difference between the true and predicted bounding box dimensions and center coordinates, and L_{mask} is the average binary cross-entropy loss.

The PointRend (PR) mask head is well-suited for refinement of detailed segmentations [11]. During inference, PR iteratively computes the mask prediction. It includes a point selection strategy which adaptively selects the most-uncertain pixels in an image that likely lie on object boundaries and interpolates their pixel values using the values of the four nearest neighbors.

III. Perception of Wires in Imagery

Previous work automatically generates semantic labels for a wire data set using chroma-key separation [16]. Each image in the data set has a corresponding segmentation mask which describes the segmentation for every wire in the scene. The approach used to automatically label these images is unable to distinguish between wire objects in a scene. The semantic segmentation frameworks which are trained on this data set are also not designed for instance-level tasks [17].

Other work on wire perception in RGB imagery performs wire segmentation without complex classification, localization, and mask prediction architectures. The Ariadne algorithm first detects wire terminals using a convolutional neural network, then performs a biased random walk over the region adjacency graph of the source image according to the color histogram and curvature likelihood of each superpixel [10]. For visually-distinct wire instances with distinct color and curvature properties, the Ariadne algorithm can distinguish individual wire instances and perform wire segmentation under occlusion.

Ariadne+ is the first method to address the wire instance segmentation problem using deep learning; it is more robust to noise than Ariadne [18]. Ariadne+ is initialized with a binary semantic segmentation mask to identify all wires in the image [16]. This mask is partitioned into superpixels to create a region adjacency graph. Each graph node is scored according to its status as an endpoint, segment, or intersection. Intersection groups are scored based on color and curvature properties, and paths are computed between candidate endpoints. At wire intersections, binary classification is used to distinguish foreground wires from background wires. Path layouts are used to resolve wire instances.

IV. Training Mask R-CNN

UIUCWires defines two new categories of segmentation annotation for objects under occlusion, namely the *Object Semantics* (OS) and the *Object Segment Semantics* (OSS) formats. When creating segmentation masks, only the portions of objects which are in view in an image receive a mask. Furthermore, the annotation for occluded objects comprises multiple contours describing the segments of the object that are in view. The OS format of segmentation annotation represents all contours of an object within one mask;

the mask captures occlusion and wire crossings. The OSS format of segmentation annotation represents one contour of an object within the mask. UIUCWires provides both the OS and OSS annotation with corresponding category and bounding box information for every object in every image, and demonstrates significant improvement in wire instance segmentation tasks when training Mask-RCNN using OSS annotations as opposed to OS annotations [13].

This work uses Mask R-CNN as implemented in Detectron2 trained on the UIUCWires data [19]. The two models tested use the R50-FPN backbone network with the PointRend (PR) mask head with each of the OS and OSS mask representation schemes. The models are initialized with Mask R-CNN baselines pretrained on COCO instance segmentation tasks. The models were each trained with a learning rate of 0.00025, 30,000 iterations, and 8 images per batch [13].

V. Wire Instance Segmentation from RGBD Imagery

The complete instance segmentation procedure which generates object instance predictions on an RGB image and uses the instance bitmasks to segment the corresponding registered depth image to produce an object instance-aware point cloud is outlined in Figure 1. The proposed method of point cloud segmentation using the PR+OSS model for segmentation produces significantly more complete segmented point clouds than the PR+OS model as shown in in Figure 2. Preliminary results on two test scenes indicate an **153%** improvement in the number of points recovered in depth segmentation using the PR+OSS model for segmentation as compared to the PR+OS model. These results are summarized in Table I.

The segmented point cloud was used to construct an object mesh of the scene using the Vedo 3D visualization library [20]. The mesh was processed with an outlier removal step using density-based clustering. The smoothed mesh is shown in Figure 3. The mesh reconstruction is a candidate 3D representation of the scene from raw RGB-D data. This 3D information about the shape of wires in the scene can be used to generate the normal vectors at each point along the surface of the mesh for use in grasp planning.

VI. Conclusions and Future Work

This work contributes an object instance-aware method for extracting unique wire and device point clouds from RGBD imagery by using Mask R-CNN trained on UIUCWires to segment and process a depth image. The OSS+PR model produces the most complete point clouds, but self-occlusion still confounds wire point cloud segmentation. This work leaves three questions unanswered for future work:

1) Can a graph search method, such as the method used in Ariadne+, be combined with the instance segmentation performed by Mask R-CNN to improve instance segmentation performance for the class of wire instance segmentation problems?
2) Can multiple fused sensor perspectives improve scene point cloud and mesh retrieval for the 3D wire instance segmentation problem?
3) How do wire detection and segmentation frameworks trained on [13] and [16] compare?

TABLE I

Number of Points in Instance-Segmented Point Clouds

Training Configuration	Two-Wire Scene	Three-Wire Scene
PR+OSS	21,688	26,366
PR+OS	8,580	4,082

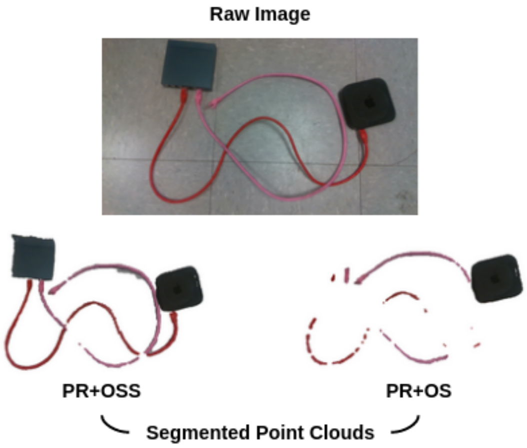

Fig. 2. Mask R-CNN with the PointRend (PR) mask head trained on the UIUCWires dataset with Object Segment Semantics (OSS) annotations produces more complete point cloud segmentations than the same model trained on the same dataset with Object Semantics (OS) annotations.

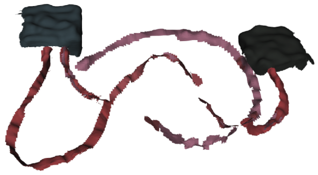

Fig. 3. The point cloud segmented using the PointRend (PR) Object Segment Semantics (OSS) training configuration can be converted to an object mesh for object grasping and manipulation applications.

The goal of this work is to demonstrate a method for 3D perception of wires to use for robotic interaction in a greater variety of contexts. In addressing remaining open questions, future work aims to improve the reconstruction quality for model-free deformable object perception.

Acknowledgements

The authors thank Hameed Abdul-Rashid and Diego Rojas for engaging in helpful discourse about this work. Part of this work was performed during Jingyi Xiang's internship with the Undergraduate Research Apprenticeship Program and during Harry Zhao's internship through the Illinois Space Grant Consortium Undergraduate Research Opportunity Program. This work was supported in part by the NASA Space Technology Graduate Research Opportunity award 80NSSC21K1292 and the U.S. Department of Education through the Graduate Assistance in Areas of National Need Fellowship Program award P200A180050-19 at the University of Illinois Urbana-Champaign Department of Aerospace Engineering.

REFERENCES

[1] M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, and K. Goldberg, “Segmenting Unknown 3D Objects from Real Depth Images Using Mask R-CNN Trained on Synthetic Data,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 7283–7290.

[2] M. Yan, Y. Zhu, N. Jin, and J. Bohg, “Self-Supervised Learning of State Estimation for Manipulating Deformable Linear Objects,” in *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, Apr. 2020, pp. 2372–2379.

[3] O. Roussel, A. Borum, M. Taäx, and T. Bretl, “Manipulation Planning with Contacts for an Extensible Elastic Rod by Sampling on the Submanifold of Static Equilibrium Conditions,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 3116–3121.

[4] A. Sintov, S. Macenski, A. Borum, and T. Bretl, “Motion Planning for Dual-Arm Manipulation of Elastic Rods,” in *IEEE Robot. Autom. Lett.*, Oct. 2020, pp. 6065–6072.

[5] S. Javdani, S. Tandon, J. Tang, J. O’Brien, and P. Abbeel, “Modeling and Perception of Deformable One-Dimensional Objects,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2011, pp. 1607–1614.

[6] A. Myronenko and X. Song, “Point Set Registration: Coherent Point Drift,” in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, Dec. 2010, pp. 2262–2275.

[7] C. Chi and C. Berenson, “Occlusion-Robust Deformable Object Tracking Without Physics Simulation,” in *IEEE/RSJ Int. Conf. Intell. Robot. Sys. (IROS)*, 2019.

[8] Y. Wang, D. McConachie, and D. Berenson, “Tracking Partially-Occluded Deformable Objects while Enforcing Geometric Constraints,” in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021.

[9] A. Caporali, K. Galassi, G. Laudante, G. Palli, , and S. Pirozzi, “Combining Vision and Tactile Data for Cable Grasping,” in *IEEE/ASME Int. Conf. Adv. Intell. Mech. (AIM)*. IEEE, July 2021, pp. 436–441.

[10] D. De Gregorio, G. Palli, and L. Di Stefano, “Let’s Take a Walk on Superpixels Graphs: Deformable Linear Objects Segmentation and Model Estimation,” in *Asian Conf. Comput. Vis. (ACCV)*. Springer, Dec. 2018, pp. 662–667.

[11] A. Kirillov, Y. Wu, K. He, and R. Girshick, “PointRend: Image Segmentation as Rendering,” in *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2020, pp. 9796–9805.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Oct. 2017, pp. 2980–2988.

[13] H. Dinkel, H. Zhao, J. Xiang, B. Coltin, T. Smith, and T. Bretl, “Benchmarking Wire Instance Segmentation with Mask R-CNN,” in *Under Submission*, 2022.

[14] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T. Lin, E. Cubuk, Q. Le, and B. Zoph, “Simple Copy-Paste is a Strong Data Augmentation Method for Instance Segmentation,” in *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*. IEEE, June 2021, pp. 2918–2928.

[15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2016, pp. 770–778.

[16] R. Zanella, A. Caporali, K. Tadaka, D. De Gregorio, and G. Palli, “Auto-Generated Wires Dataset for Semantic Segmentation with Domain Independence,” in *IEEE Int. Conf. Comput. Cont. Robot. (ICCCR)*. IEEE, Jan. 2021, pp. 292–298.

[17] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation,” in *Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 801–818.

[18] A. Caporali, R. Zanella, D. De Gregorio, and G. Palli, “Ariadne+: Deep Learning-based Augmented Framework for the Instance Segmentation of Wires,” in *IEEE Trans. Ind. Inf.*, February 2022, pp. 1–11.

[19] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, “Detectron2,” https://github.com/facebookresearch/detectron2, 2019.

[20] M. Musy, G. Jacquenot, G. Dalmasso, neoglez, R. de Bruin, A. Pollack, F. Claudi, C. Badger, icemtel, Z.-Q. Zhou, B. Sullivan, B. Lerner, D. Hrisca, and D. Volpatto, “Vedo: A Python Module for Scientific Analysis and Visualization of 3D Objects and Point Clouds,” https://vedo.embl.es/, 2022.